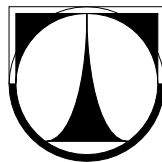


**TECHNICKÁ UNIVERZITA V LIBERCI**  
Fakulta mechatroniky, informatiky a mezioborových studií



## BAKALÁŘSKÁ PRÁCE

Liberec 2012

**Milan Eichler**

**TECHNICKÁ UNIVERZITA V LIBERCI**  
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B2612 – Elektrotechnika a informatika  
Studijní obor: Informatika a logistika

**Detekce polohy částic z digitálních  
hologramů**

**Particle detection from digital  
holograms**

**BAKALÁŘSKÁ PRÁCE**

Autor:	<b>Milan Eichler</b>
Vedoucí práce:	Ing. Pavel Psota
Konzultant:	Ing. Roman Doleček

**V Liberci 2.1. 2012**

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Milan Eichler**  
Osobní číslo: **M09000012**  
Studijní program: **B2612 Elektrotechnika a informatika**  
Studijní obor: **Informatika a logistika**  
Název tématu: **Detekce poloh částic z digitálních hologramů**  
Zadávající katedra: **Ústav řízení systémů a spolehlivosti**

### **Z á s a d y   p r o   v y p r a c o v á n í :**

1. Prostudujte odbornou literaturu a vypracujte literární rešerši pojednávající základní principy holografie, PIV a holografické PIV.
2. Na základě rešerše navrhnete holografické uspořádání pro detekci poloh částic.
3. Navržené uspořádání sestavte a proveďte jeho funkčnost.
4. Zpracujte a zhodnoťte dosažené výsledky.

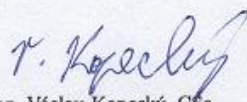


Rozsah grafických prací: dle potřeby  
Rozsah pracovní zprávy: cca 40 stran  
Forma zpracování bakalářské práce: tištěná/elektronická  
Seznam odborné literatury:


1. KREIS, T. Handbook of Holographic Interferometry : Optical and Digital Methods . Berlin : Wiley, 2004. 542 p.
2. SALEH, B. E. A; TEICH, M. C. Základy fotoniky. Praha : MATFYZPRESS, 1994. 1056 p.

Vedoucí bakalářské práce: **Ing. Pavel Psota**  
Ústav řízení systémů a spolehlivosti  
Konzultant bakalářské práce: **Ing. Roman Doleček**  
Ústav řízení systémů a spolehlivosti

Datum zadání bakalářské práce: **14. října 2011**  
Termín odevzdání bakalářské práce: **18. května 2012**

  
prof. Ing. Václav Kopecký, CSc.  
děkan



  
doc. Ing. Libor Tůma, CSc.  
vedoucí ústavu

V Liberci dne 14. října 2011

### **Prohlášení**

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum

3. 7. 2012

Podpis

*Giebler*

## **Poděkování**

Na tomto místě bych rád poděkoval především vedoucímu mé práce ing. Pavlovi Psotovi za odborné vedení, potřebnou kritiku i za poskytování cenných rad a připomínek. Dále bych rád poděkoval všem ostatním lidem, kteří mě podporovali a pomáhali mi.

## **Abstrakt**

Bakalářská práce se zabývá návrhem a realizací optického systému pro detekování polohy částic v kapalině pomocí digitálních hologramů. Dále obsahuje návržení a realizaci softwaru v prostředí MATLAB tak, aby bylo možné využít existujícího softwaru pro rekonstrukci digitálních hologramů, pro vyhodnocení a zobrazení naměřených dat. V závěru práce jsou názorně zobrazeny softwarem detekované částice a shrnuty výhody a nevýhody použitého optického uspořádání a navrženého softwaru.

**Klíčová slova:** Digitální holografie, PIV, laser, MATLAB, Holografická interferometrie

## **Abstract**

This thesis focuses on design and implementation of an optical system for detecting the position of particles in liquid by using digital holograms. It further includes designing and implementing software in MATLAB so it would be possible existing software for the reconstruction of digital holograms to evaluation and display of measured data. At the end of the work illustrative examples of software detected particles are displayed and the advantages and disadvantages of the proposed optical arrangement and designed software are summarized.

**Keywords:** Digital holography, PIV, laser, MATLAB, Holographic interferometry



## Obsah

Poděkování.....	4
Abstrakt.....	5
Abstract.....	5
Obsah .....	6
Seznam obrázků .....	8
Seznam zkratk .....	9
Úvod.....	10
1 Teoretická část .....	11
1.1 Světlo.....	11
1.1.1 Intenzita .....	12
1.1.2 Odraz a lom světla .....	13
1.1.3 Polarizace.....	14
1.1.4 Interference .....	18
1.1.5 Difrakce .....	20
1.2 Laser.....	22
1.2.1 Princip laseru .....	22
1.2.2 Typy laseru .....	23
1.3 Holografie .....	24
1.3.1 Záznam hologramu .....	24
1.3.2 Rekonstrukce hologramu .....	26
1.3.3 Digitální holografie.....	26
1.3.4 Holografická interferometrie .....	28
1.4 PIV .....	28
1.4.1 Princip PIV .....	29
1.5 HPIV .....	29
2 Experimentální část.....	32
2.1 Popis experimentu.....	32
2.2 Použité Komponenty .....	34
2.3 Rozvržení jednotlivých komponent .....	33
2.4 Softwarová implementace.....	39
2.4.1 Princip softwaru .....	39
3 Výsledky experimentu .....	43
4 Závěr .....	47
Seznam použité literatury .....	49
Příloha.....	50
Příloha 1 – Program CircularHough_Grd .....	50



Příloha 2 – Program Pro detekci polohy částic .....	59
---	----

## Seznam obrázků

Obr 1. Barevné spektrum .....	11
Obr 2. Odraz a lom světla.....	13
Obr 3. Šíření světla .....	14
Obr 4. Druhy polarizace. a) Eliptická polarizace, b) Kruhová polarizace, c) lineární polarizace.....	15
Obr 5. Polarizace odrazem.....	16
Obr 6. Polarizace absorpcí.....	17
Obr 7. Youngův pokus.....	19
Obr 8. Interferenční obrazec .....	19
Obr 9. Mýdlová bublina.....	20
Obr 10. Difrakce na štěrbině .....	21
Obr 11. Fresnelova a Fraunhoferova zony .....	21
Obr 12. Schéma funkce laseru .....	23
Obr 13. Záznam hologramu.....	25
Obr 14. Rekonstrukce Hologramu.....	26
Obr 15. Geometrie digitální Fresnelovy holografie .....	27
Obr 16. Princip PIV.....	29
Obr 17. Základní uspořádání in-line holografie .....	30
Obr 18. Základní uspořádání tomografické holografie.....	31
Obr 19. Digitální hologram .....	32
Obr 20. Schématické znázornění rozvržení jednotlivých komponent .....	33
Obr 21. Reálné rozložení při experimentu.....	34
Obr 22. AVT kamera Stigra F-504(převzato z [6]) .....	34
Obr 23. Laser Compass 315M-150SL .....	35
Obr 24. Prostorový filtr, a)mikrometrické šrouby pro posuv clonky v horizontálním a vertikálním směru, b)mikroposuv objektivu ve směru clonky .....	36
Obr 25. Kolimační objektiv.....	37
Obr 26. Částice 0,5 mm .....	38
Obr 27. Identifikace částic.....	40
Obr 28. Postupné zaostřování detekované částice v postupných vzdálenostech od CCD, $a=30,02$ cm, $b=30,04$ cm, $c=30,06$ cm, $d=30,08$ cm, $e=31$ cm, $f=31,02$ cm, $g=31,04$ cm, $h=31,06$ cm, $i=31,08$ cm, $j=32$ cm, $k=32,02$ cm, $l=32,04$ cm, $m=32,06$ cm, $n=32,08$ cm, $o=33$ cm . - Označený obrázek byl zvolen jako nejostřejší .....	42
Obr 29. Zobrazení pozice částic ve 3D mapě .....	43
Obr 30. Zobrazení pozice částic v rovině X-Y .....	44
Obr 31. Zobrazení pozice částic v rovině X-Z .....	44
Obr 32. Zobrazení pozice částic v rovině Y-Z .....	45
Obr 33. Reálné rozmístění částic v nádobě .....	45

## Seznam zkratk

Značka	Rozměr	Význam
$c$	$[m.s^{-1}]$	Rychlost světla
$B$	$[kg.s^{-1}.C^{-1}]$	Magnetická indukce
$E$	$[J]$	Energie
$\mathbf{E}$	$[V.s^{-1}]$	Vektor intenzity elektrického pole
$E_O, E_R$		Reálná amplituda předmětové a referenční vlny
$f$	$[Hz]$	Frekvence
$\mathbf{H}$	$[A.m^{-1}]$	Vektor intenzity magnetického pole
$h$	$[J.s]$	Planckova konstanta
$I$	$[W.m^{-2}]$	Intenzita světla
$l$	$[m]$	Koherentní délka
$n$		Index lomu prostředí
$\mathbf{S}$	$[W.m^{-2}]$	Poyntingův vektor
$\Delta t$	$[s]$	Časový interval
$v$	$[m.s^{-1}]$	Rychlost šíření světla v prostředí
$\mathbf{v}$	$[m.s^{-1}]$	Rychlost částic
$\Delta \mathbf{X}$	$[m]$	Průměrný posun částic
$\alpha$	$[^\circ]$	Úhel dopadu světelného svazku
$\alpha_O, \alpha_R$		Reálná amplituda objektové a referenční vlny
$\vartheta$	$[Hz]$	Frekvence fotonu
$\lambda$	$[m]$	Vlnová délka
$\mu$	$[N.A^{-2}]$	Permeabilita vakua
$\varphi_O, \varphi_R$	$[rad]$	Reálná fáze objektové a referenční vlny

## Úvod

I v dnešní době, navzdory velkému rozmachu modelovacích technik, je reálné měření jednou z nejdůležitějších technik pro zkoumání většiny jevů. Modelovací softwary, i přes neustále se zvětšující výpočetní výkon, nedokáží naprosto přesně popsat chování jednotlivých dějů, protože i ten nejlepší model je vždy jen idealizované nahrazení skutečnosti. Z tohoto důvodu existuje neustálá potřeba zdokonalovat měřicí techniky k zachycení reálných dějů ve zkoumaných procesech.

Pro zkoumání proudění tekutin dosahují nejlepších výsledků měření především bezkontaktní metody. Jednou z nejkompexnějších je metoda laserové anemometrie Particle Image Velocimetry (PIV). Jedná se o metodu pro detekci částic ve 2D řezu v dané tekutině. Dvojitou expozicí v jedné rovině je možné určit rychlost těchto částic. Laser Doppler anemometrie (LDA), je technika umožňující měření rychlosti částic nebo kapiček, Phase Doppler anemometrie (PDA), navíc umožňuje stanovení velikosti těchto částic.

Další metoda, vhodná pro detekci částic v tekutině, je holografie. Zatímco u klasické anemometrie měříme v jediném řezu v prostoru, u holografie zaznamenáme obraz částic v celém třidimensionálním prostoru pro jednotlivé časové okamžiky.

Rozmach v polovodičovém průmyslu a vysoké rozlišení moderních digitálních senzorů umožnil vznik digitální holografie. V dřívějších dobách byly hologramy zaznamenávány na fotografické desky, foto termoplastické fólie nebo fotorefrakční krystaly. Všechny tyto metody mají specifické nevýhody. Fotografické desky nebo plechy potřebují mokré chemické zpracování, foto termoplastické fólie vyžadují časově náročné elektrické zpracování a fotorefraktivní krystaly vyžadují optické obnovení a neumožňují dlouhodobé ukládání. Oproti tomu u digitální holografie se rekonstrukce provádí numericky, odpadá tak nutnost chemického zpracování záznamu a dává nám obrovské možnosti ukládání a následného zpracování zaznamenaných dat.

Bakalářská práce se zabývá využitím digitální holografie pro detekci částic v tekutině. Práce je rozdělena do dvou částí. První z nich obsahuje teorii nezbytnou pro pochopení základních principů fungování holografie nutných k úspěšnému návrhu a sestavení měřicí soustavy. Druhá část popisuje návrh a realizaci měřicí soustavy a popis naprogramovaného softwaru nutného k vyhodnocení získaných dat.

# 1 Teoretická část

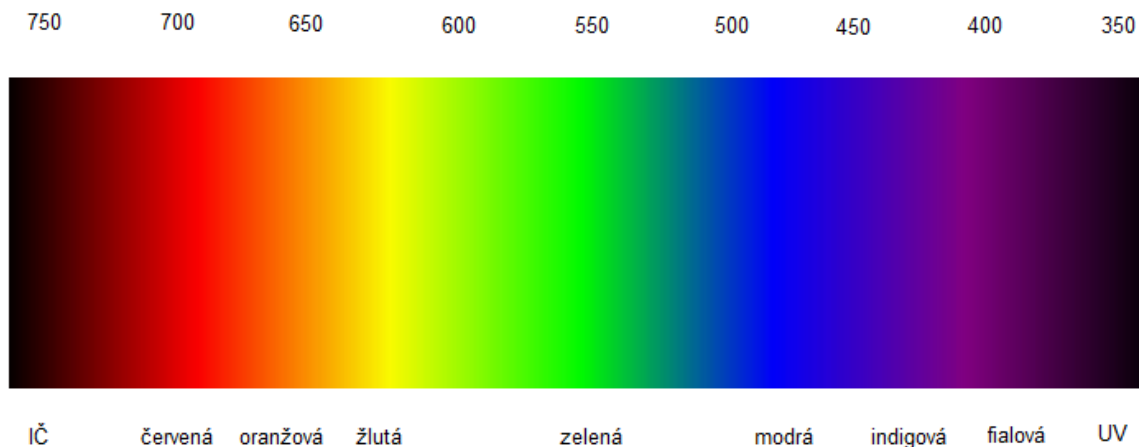
## 1.1 Světlo

Světlem nazýváme část elektromagnetického záření, které můžeme vidět lidským okem. Z celého spektra elektromagnetického záření je lidské oko schopné vnímat jen úzkou část záření o frekvencích  $3,9 \cdot 10^{14}\text{Hz}$  až  $7,7 \cdot 10^{14}\text{Hz}$ . Důležitou charakteristickou vlastností světla je jeho rychlost. Rychlost světla ve vakuu byla stanovena  $c = 299792458 \text{ m} \cdot \text{s}^{-1}$ .

V ostatních prostředí je rychlost šíření světla menší. Podobně jako ostatní vlnění i světlo charakterizuje jeho vlnová délka  $\lambda$

$$\lambda = \frac{c}{f} \quad (1)$$

Jednotlivé barvy jsou tedy v intervalu 390nm - fialová barva až 770nm - červená barva (viz. Obr 1.). Toto vymezení je jen přibližné, neboť je omezeno individuálními vlastnostmi lidského oka. Navíc každý člověk vnímá jednotlivé vlnové délky různě.



Obr 1. Barevné spektrum

Při průchodu světla prostředím se frekvence nemění, mění se však rychlost světla dle:

$$f = \frac{c}{\lambda_0} = \frac{v}{\lambda}, \quad (2)$$

kde  $\lambda_0$  je vlnová délka světla ve vakuu a  $\lambda$  je vlnová délka světla v daném prostředí. Index lomu prostředí  $n$  je definován

$$n = \frac{c}{v} = \frac{\lambda_0}{\lambda}. \quad (3)$$

### 1.1.1 Intenzita

Intenzita světla je jediný parametr, který přímo ovlivňuje světelné senzory, jako je oko nebo CCD kamera. Je definovaná jako světelný tok dopadající na určitou plochu za jednotku času. Intenzita je dána velikostí Poyntingova vektoru, který je dán vektorovým součinem intenzit elektrického a magnetického pole

$$\vec{S} = \vec{E} \times \vec{H}. \quad (4)$$

Jelikož vektory  $E$  a  $H$ , intenzity elektrického, respektive magnetického pole, jsou na sebe kolmé, můžeme pro velikost tohoto vektoru psát

$$|S| = I = |EH|. \quad (5)$$

Za intenzitu magnetického pole  $H$  dosadíme:

$$H = \frac{B}{\mu} = \frac{E}{\mu c}. \quad (6)$$

Potom dostaneme vztah

$$I = \frac{E^2}{\mu c}. \quad (7)$$

### 1.1.2 Odraz a lom světla

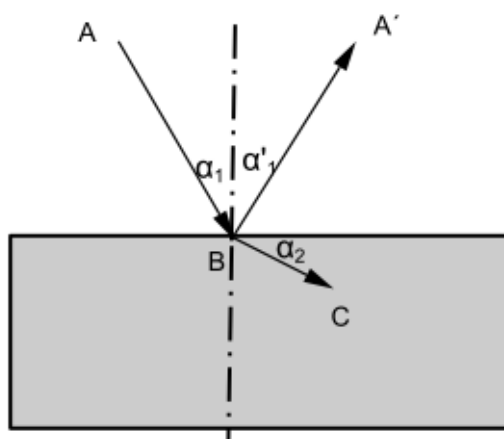
Při dopadu světla na předměty z různých materiálů dochází na rozhraní k jeho odrazu a lomu. Pro odražený paprsek bude platit vztah

$$\alpha_1 = \alpha'_1, \quad (8)$$

kde  $\alpha_1$  je úhel dopadajícího světelného svazku a  $\alpha'_1$  je úhel odraženého světelného svazku. Lom světelného svazku v prostředí popisuje Snellův princip

$$n_1 \cdot \sin \alpha_1 = n_2 \cdot \sin \alpha_2, \quad (9)$$

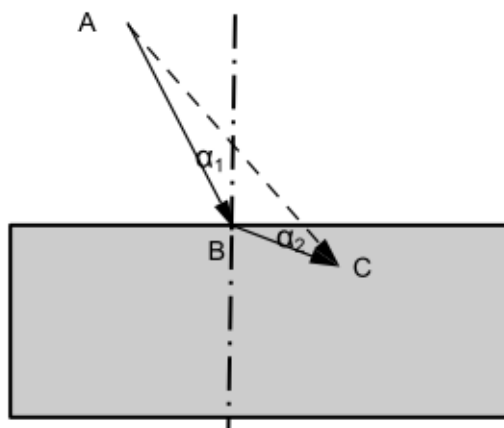
kde  $\alpha_1$  a  $\alpha_2$  jsou úhly dopadu a světelného svazku a lomu a  $n_1$  a  $n_2$  jsou indexy lomu jednotlivých prostředí.



Obr 2. Odraz a lom světla

O zevšeobecnění základních zákonů geometrické optiky se postaral Pierre de Fermat, když tyto zákony shrnul do jedné věty, tzv. Fermatův princip: Světlo se v prostoru šíří z jednoho bodu do druhého po takové dráze, aby doba potřebná k proběhnutí této dráhy nabývala extrémních hodnot vzhledem k ostatním dráhám. Extrém může dosahovat minima i maxima, většinou jde ale o minimum. V takovém případě platí, že světelný paprsek se šíří podél dráhy s nejmenší dobou šíření. Nejkratších cest může být několik, v takovém případě se světelný paprsek šíří všemi těmito cestami.





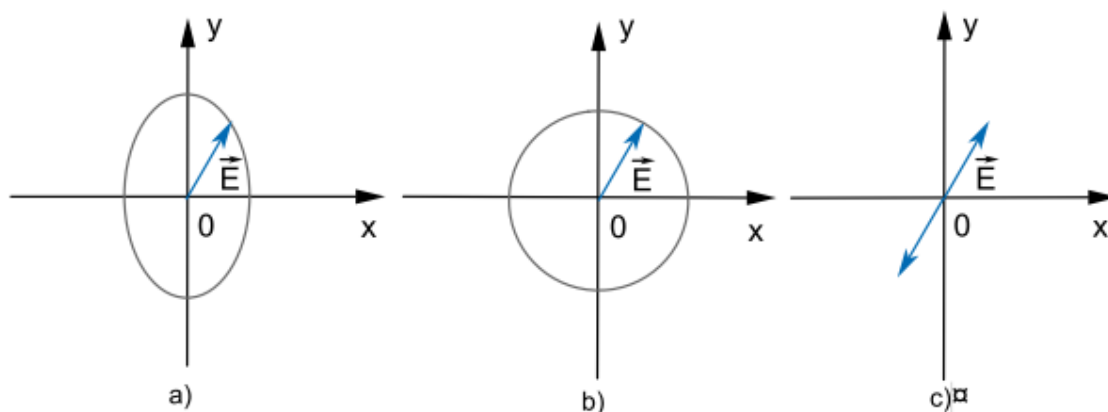
Obr 3. Šíření světla

Fermatův princip je možné využít při odvození zákona odrazu a lomu světla. Neboť šíří-li se světelný svazek z bodu A do bodu B (viz. Obr 2.) po nejrychlejší dráze a tato dráha svírá s osou kolmou na rozhraní, o které se světlo láme, úhel  $\alpha_1$ , potom šíření světelného svazku z bodu B do bodu A', který je odražen od rozhraní a je ve stejném prostředí, bude probíhat po stejné dráze jako z bodu A do bodu B. Z toho plyne, že úhel  $\alpha_1$  dopadajícího světelného svazku a úhel  $\alpha_2$ , odraženého svazku se musí rovnat.

Při šíření světelného svazku z bodu A do bodu C (viz. Obr 3.), dochází na rozhraní v bodě B k lomu světelného svazku tak, aby šíření světelného svazku probíhalo podle Fermatova principu, v co nejkratším čase. Proto, jedná-li se o dvě prostředí s rozdílným indexem lomu, hovoříme o lomu světla. Úhel, pod kterým se bude světelný svazek na rozhraní lámat, je dán Snellovým principem viz. výše.

### 1.1.3 Polarizace

Polarizace je vlastnost elektromagnetického vlnění, která říká, jak se mění vektor intenzity elektrického pole v čase. Říkáme, že světlo je lineárně nebo také rovinně polarizované, osciluje-li vektor intenzity elektromagnetického pole podél přímky (viz. Obr 4.c). Pohybuje-li se konec vektoru intenzity elektrického pole po elipse, říkáme, že světlo je elipticky polarizované (viz. Obr 4.a). Pohybuje-li se konec intenzity elektrického pole po kružnici, dostáváme kruhovou polarizaci (viz. Obr 4.b). Pokud bychom se dívali přímo proti letícímu světlu a vektor intenzity elektrického pole by se otáčel proti směru hodinových ručiček, šlo by o pravotočivou polarizaci, v opačném případě bychom měli levotočivou polarizaci.



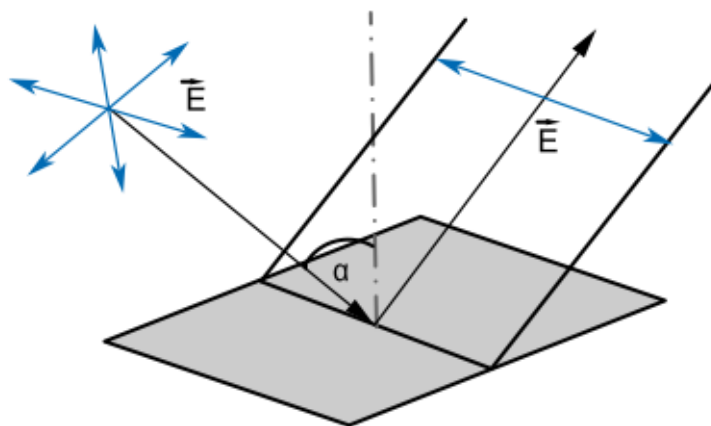
Obr 4. Druhy polarizace. a) Eliptická polarizace, b) Kruhová polarizace, c) lineární polarizace

Tímto jsme vyčerpali všechny možné druhy polarizovaného světla. Existuje však ještě nepolarizované světlo. Tento případ nastává, není-li světlo dokonale monochromatické, nebo když poměry fází  $x$  a  $y$  nejsou dokonale ustálené. V tomto případě vektory intenzity elektrického pole nejprve kmitají v jednom směru a posléze v opačném.

V praxi má největší význam polarizace lineární, a proto se často uvádí jako polarizované elektromagnetické vlnění pouze elektromagnetické vlnění, polarizované lineárně. I nás bude nejvíce zajímat, jak donutit vektor intenzity elektrického pole kmitat pouze v jednom směru, tedy jak docílit aby byl vektor lineárně polarizovaný. K tomuto účelu nejčastěji používáme tři základní způsoby, polarizaci odrazem, dvojlomem a absorpcí.

### Polarizace odrazem

Předpokládejme, že rozhraní dvou materiálů dopadá nepolarizované světlo pod úhlem dopadu  $\alpha$  a odráží se pod úhlem  $\alpha'$ , za určitých okolností (průhledných rozhraní nebo průsvitných prostředí, vhodný úhel dopadu) se může také lámat do druhého prostředí. Odražený paprsek bude částečně lineárně polarizovaný a jeho vektor intenzity elektrického pole bude kmitat v rovině kolmé na rovinu dopadu (viz. Obr 5). Stupeň polarizace závisí na úhlu dopadu.



Obr 5. Polarizace odrazem

Bude-li světelná vlna dopadat pod tzv. Brewsterovým úhlem  $\alpha$ , nastane při odrazu světla úplná lineární polarizace a odražený paprsek bude úplně lineárně polarizován. Velikost Brewsterova úhlu závisí na indexu lomu  $n_2$  a  $n_1$  obou prostředí, na jejichž rozhraní dochází k odrazu světla. Pro jeho velikost platí vztah

$$n_1 \cdot \tan \alpha = n_2. \quad (10)$$

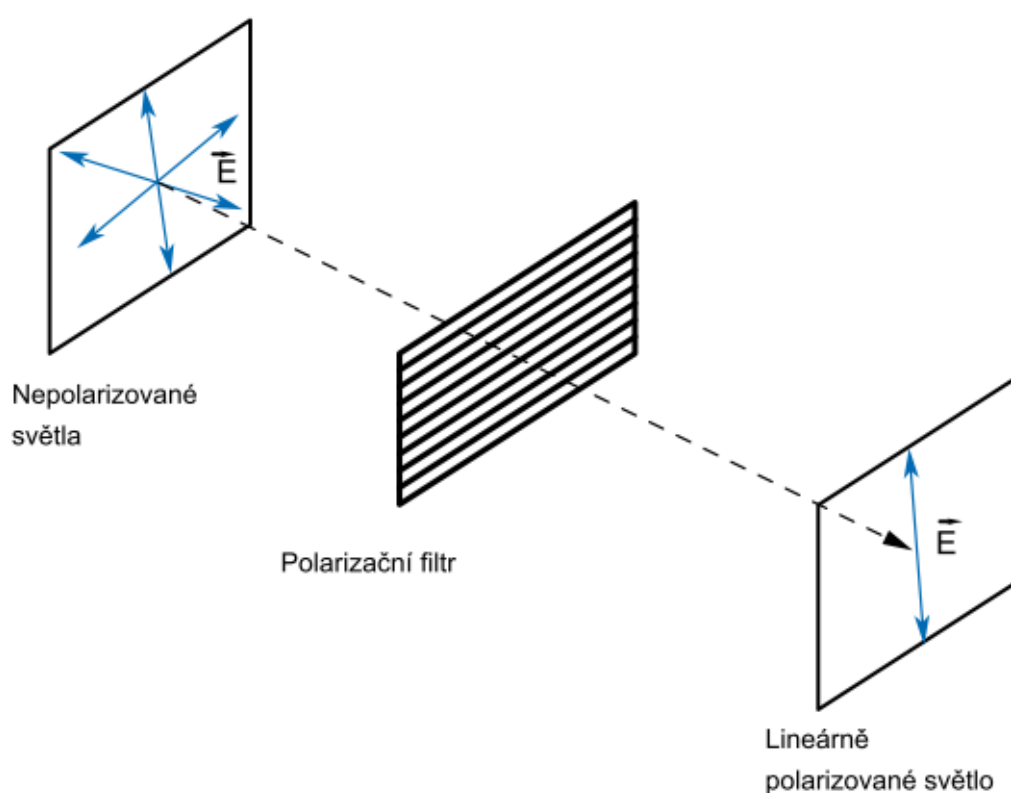
### Polarizace dvojlomem

K polarizaci dvojlomem dochází, dopadá-li světlo na látku, která má různé indexy lomu pro světlo lineárně polarizované v jednom směru a světlo lineárně polarizované v druhém směru. K tomuto dochází u látek, které se skládají z dlouhých nesférických molekul, jejichž podélná osa je výrazně delší než příčná a jsou v látce uloženy rovnoměrně. Mezi takové látky patří například islandský vápenec a křemen.

Dopadá-li na takovou látku nepolarizované světlo, rozdělí se při průchodu na dva paprsky, řádný (řídí se Snellovým zákonem lomu a má konstantní index lomu) a mimořádný (neřídí se Snellovým zákonem lomu, jeho index lomu závisí na směru, v němž se světlo krystalem šíří). Říkáme, že nastal dvojlom. Oba paprsky jsou úplně lineárně polarizované a jejich intenzity elektrického pole kmitají v navzájem kolmých kmitových rovinách.

## Polarizace absorpcí

Některé krystaly se vyznačují tím, že propouští pouze světelný svazek, který kmitá v daném směru. Můžeme si je představit jako materiál složený z dlouhých tenkých štěrbin. Projít může pouze to vlnění, jehož elektrická složka je kolmá na tyto štěrby, ostatní se pohltí. Polarizované paprsky vzniklé v anizotropním krystalu mohou být absorbovány s různou intenzitou. Pokud je jeden svazek téměř absorbován a druhý krystalem prochází jen s minimálním zeslabením intenzity, dostáváme za krystalem lineárně polarizované světlo. Tohoto jevu se využívá u polarizačních filtrů, které propouštějí pouze světelný svazek polarizovaný v daném směru.



Obr 6. Polarizace absorpcí

### 1.1.4 Interference

Interference označuje vzájemné ovlivňování, prolínání nebo střetávání jevů či hmoty. Jedná se také o charakteristickou vlastnost vln. Jestliže se v jednom prostředí šíří vlnění z více různých zdrojů, šíří se tak, že se tato vlnění navzájem neovlivňují. Pakliže tato vlnění přicházejí do určitého bodu, dochází k jejich skládání. U mechanického vlnění dochází ke sčítání jejich okamžité výchylky, u elektromagnetického vlnění se sčítají hodnoty jejich elektrické a magnetické složky.

Vzájemným skládáním světelných vln se střídavě objevují tmavé a světlé proužky (u monochromatického světla). Tento jev skládání svazků poprvé vysvětlil Thomas Young (viz Obr 7), jehož pokus popisuje světlo vycházející z bodového zdroje a dopadající na dvojici štěrbin.

Jako zdroj světla zde slouží osvětlená štěrbina  $Z$ . Tato štěrbina, se chová jako bodový zdroj světla, takže se světlo bude šířit také do prostoru za překážkou a bude osvětlovat další dvě štěrby  $Z_1$  a  $Z_2$ . Tyto štěrby se budou opět chovat jako bodové zdroje světla a budou osvětlovat stínítko  $S$ . Podle zákonů paprskové optiky (konkrétně podle zákona přímočarého šíření světla) světlo nemůže projít přes druhou dvojici štěrbin a nemůže dopadnout na stínítko. Ve skutečnosti se na stínítku objeví soustava světlých a tmavých proužků – interferenční obrazec, interferogram (viz Obr 8), což je důkazem vlnových vlastností světla.

To jestli na stínítku vzniknou světlé nebo tmavé proužky, je dáno rovnicí pro extrémní podmínky. Tam, kde vidíme světlý proužek, vzniká interferenční maximum, naopak v místě vzniku tmavého proužku je interferenční minimum.

Rovnice pro maximum v interferenčním poli je dána vztahem

$$\Delta l = 2 \cdot k \cdot \frac{\lambda}{2}, \quad (11)$$

kde  $k$  je řád interferenčního maxima a nabývá hodnot  $0,1,2,3,\dots$  a  $\lambda$  je vlnová délka světla.

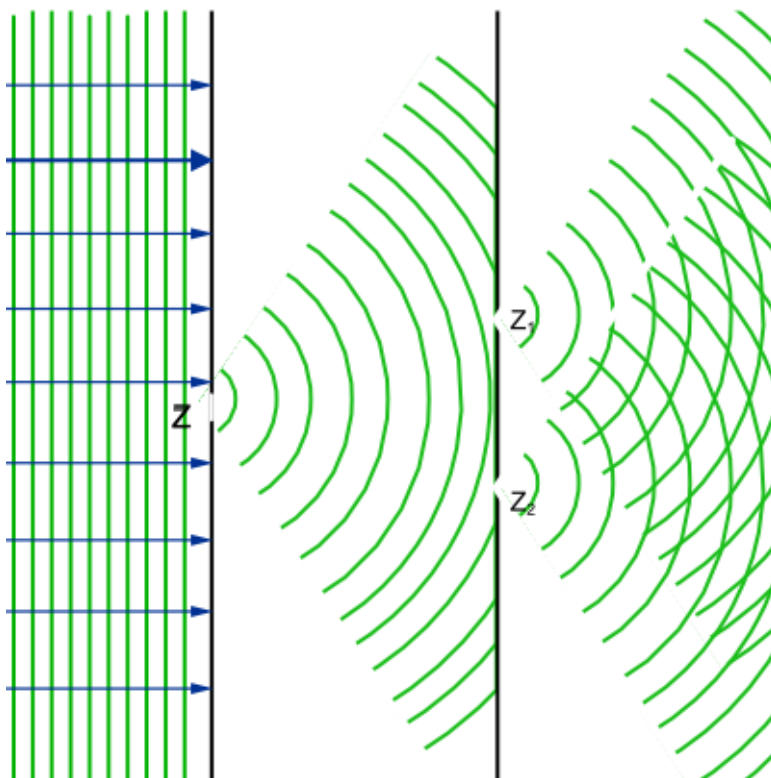
Rovnice pro minimum v interferenčním poli je dána vztahem

$$\Delta l = (2 \cdot k - 1) \cdot \frac{\lambda}{2}, \quad (12)$$

kde  $k$  je řád interferenčního minima a nabývá hodnot  $0,1,2,3\dots$  a  $\lambda$  je vlnová délka světla. Aby tento interferenční obrazec vůbec mohl vzniknout, musí záření splňovat určité podmínky

- 1) všechna záření dopadající do jednoho bodu na stínítku musí mít stejnou vlnovou délku;
- 2) v daném bodě na stínítku musí mít všechna záření stálý, s časem neměnný, dráhový rozdíl (tzn. také stálý fázový rozdíl).

Záření, která splňují obě podmínky, označujeme jako koherentní záření. Za koherentní můžeme považovat záření, která prochází štěrbinami  $Z_1$  a  $Z_2$  v případě, že je jejich vzdálenost velmi malá. Dnes už můžeme využít dalšího zdroje koherentního záření, kterým je laserové záření.



Obr 7. Youngův pokus



Obr 8. Interferenční obrazec

Pokud se dráhový rozdíl interferenčních svazků blíží nule, můžeme všechny paprsky považovat za koherentní a budou spolu navzájem interferovat.

K tomu dochází například na tenké vrstvě materiálu, jako je tenké sklo nebo mýdlová bublina (viz. Obr 9). Při dopadu paprsku na tenkostěnný materiál dochází

k odrazu paprsku od horního i dolního prostředí a může dojít ke skládání vln světla, potom bude docházet k zesílení a zeslabení určité barvy ve spektru dopadajícího bílého světla. Budeme-li uvažovat vlnu dopadající na tenkostěnný materiál ze vzduchu, vznikne rozhraní vzduch-materiál, na kterém se bude paprsek částečně odrážet a částečně procházet do druhého prostředí, kde opět dopadá na rozhraní materiál-vzduch a dochází znovu k odrazu paprsku zpět do materiálu a průchodu paprsku ven. Paprsek odražený zpět do materiálu bude interferovat s původním paprskem a dojde tak ke vzniku interferenčního obrazce.

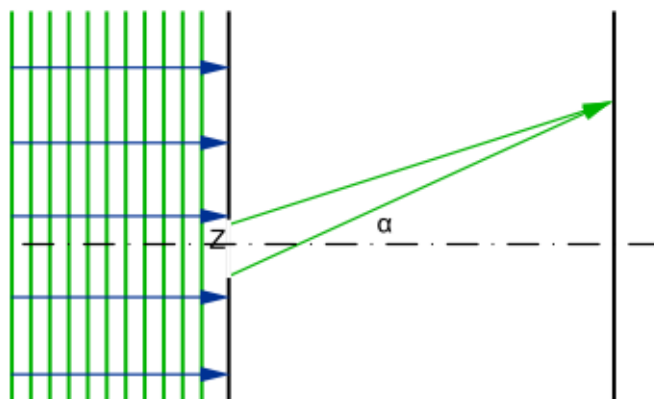


*Obr 9. Mýdlová bublina*

### **1.1.5 Difrakce**

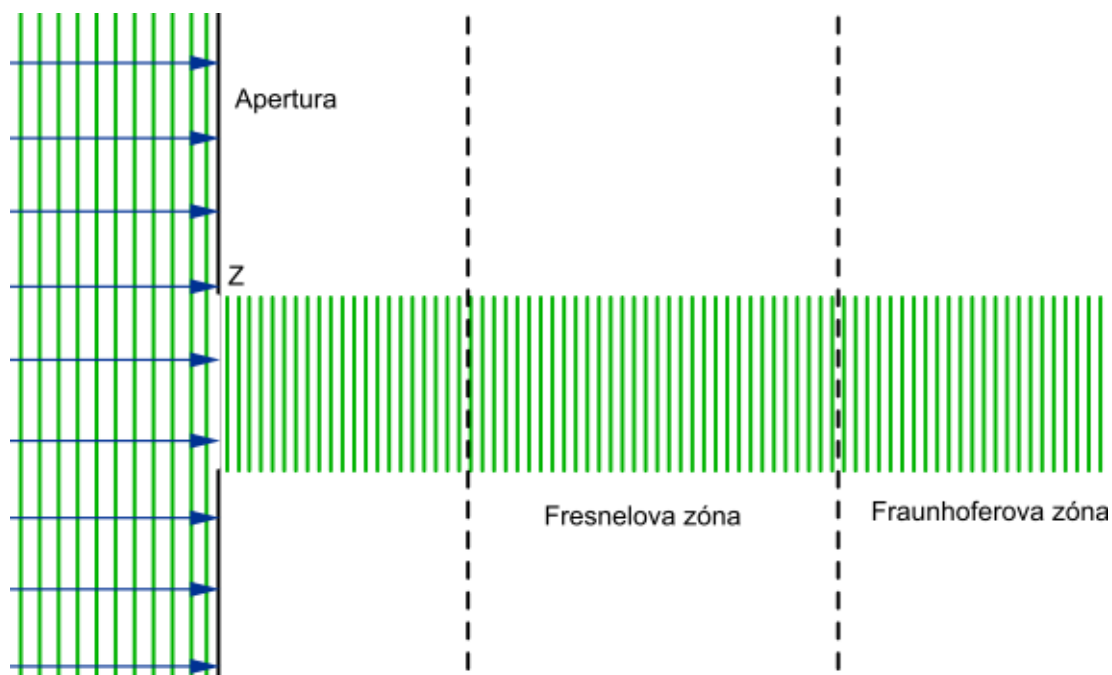
Difrakce neboli ohyb světla je jev, ke kterému dochází při dopadu světelného svazku na překážky, které jsou srovnatelné s jeho vlnovou délkou (viz. Obr 10.). Tento jev se projevuje tím, že se světlo šíří částečně i do prostoru za překážkou, kam by se podle paprskové optiky nikdy šířit nemělo, tzn. světlo se šíří i do oblasti geometrického stínu. Hranice mezi světlem a stínem potom není ostrá a na stínítku za překážkou se vytváří takzvaný difrakční obrazec. Podobně jako interferenční obrazec jej tvoří soustava nestejně širokých světlých a tmavých proužků. Tento obrazec můžeme považovat za výsledek interference světla, které do uvažovaného místa na stínítku dopadají s různým dráhovým rozdílem.





Obr 10. Difrakce na štěrbině

Difrakční jevy dělíme na Fresnelovy a Fraunhoferovy. Fresnelovy difrakční jevy jsou obecnější a matematické řešení je komplikované, ale přesnější. Fresnelova difrakce předpokládá kulové vlnoplochy jak na apertuře, tak i na stínítku, je vhodnější pro blízké vzdálenosti apertury a stínítka (viz. Obr 11.). Jelikož se Fresnelův princip uplatňuje při rekonstrukci hologramu, bude podrobněji probrán v kapitole 1.3.3. Fraunhoferova difrakce předpokládá rovinou vlnoplochu jak na apertuře otvoru, tak na stínítku. Matematický popis je jednodušší, ale platný jen v případě, kdy zdroj, apertura i stínítko jsou vůči sobě velmi vzdáleny (viz. Obr 11.).



Obr 11. Fresnelova a Fraunhoferova zony

## 1.2 Laser

Slovo laser je akronym anglického Light Amplification by Stimulated Emission of Radiation což lze volně přeložit jako zesilování světla pomocí stimulované emise záření. Jde o optický zdroj elektromagnetického záření tj. světla v širším smyslu. Laser je generátor koherentního záření s extrémní monochromaticností, tj. všechny fotony tohoto světla mají stejnou barvu, respektive vlnovou délku a fázi.

Další důležitou vlastností laseru je jeho nízká divergence, čili rozbíhavost, to znamená, že všechny fotony laserového záření se pohybují stejným směrem. Poslední důležitou vlastností laseru je vysoká hustota přenášeného výkonu či energie. Žádné jiné záření, než záření vyvolané laserem tyto vlastnosti nemá.

### 1.2.1 Princip laseru

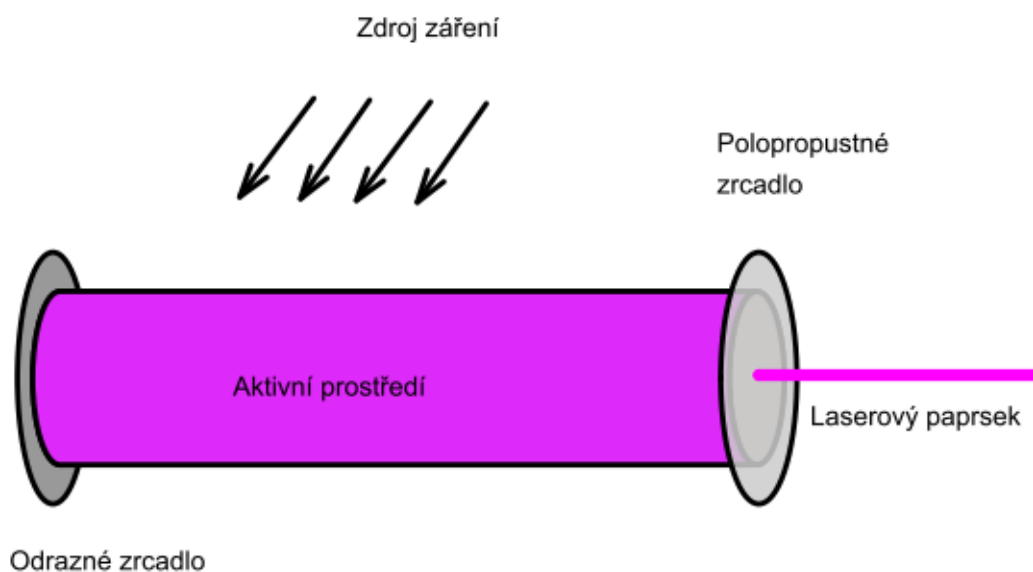
Ačkoli laser už existuje více než 40 let, princip funkce laseru je pořád stejný. Každý laser se musí skládat ze tří základních částí. Je to aktivní prostředí, kde probíhá zesílení záření zdroj čerpání pro excitaci aktivního prostředí a resonátor, který vytváří zpětnou vazbu mezi aktivním prostředím a zářením vedoucí ke vzniku laserového záření.

Největší význam pro funkci laseru má proces emise kvantové soustavy, k emisi může docházet spontánně nebo prostřednictvím stimulace. Při spontánní emisi přejde náhodně některá kvantová soustava na nižší energetickou hladinu a přitom je uvolněna excitační energie ve formě fotonu s energií

$$h\nu = E_2 - E_1, \quad (13)$$

Kde  $h$  je Planckova konstanta  $E_2 - E_1$  je rozdíl dvou energetických hladin a  $\nu$  je frekvence fotonu který odpovídá rozdílu těchto dvou hladin. Ke stimulové emisi může dojít při interakci excitovaného kvantového systému s elektromagnetickým zářením. Tohoto jevu se v laseru využívá tak, že v aktivním prostředí se nachází dostatek soustav v excitovaném stavu. Následně u některé z těchto soustav dojde ke spontánní emisi a uvolněné kvantum bude stimulovat další soustavy. U těchto soustav dojde opět k emisi a tím se optické záření bude řetězovou reakcí zesilovat.

Spontánní emise se stane zanedbatelnou oproti emisi stimulované a díky tomu nabude záření jednotné povahy. To znamená, že laserové záření bude monochromatické a koherentní. Aby docházelo k šíření laserového záření jedním směrem, tvaruje se aktivní prostředí do tvaru válce (viz. Obr 12).



Obr 12. Schéma funkce laseru

### 1.2.2 Typy laseru

Existuje několik základních typů laserů, jejichž princip funkce je sice stejný, liší se však v typu použitého aktivního prostředí. Podle typu aktivního prostředí můžeme lasery dělit na pevnolátkové, kapalinové, plynné a polovodičové lasery.

Druhým způsobem dělení laserů je podle režimu generace, tyto lasery dělíme na kontinuální a pulzní. Dalším typem dělení může být dělení podle způsobu čerpání na opticky čerpané, lasery čerpané elektrickým výbojem, chemickou reakcí, elektronovým svazkem. Lasery můžeme dělit ještě do mnoha jiných skupin, ale tyto jsou nejdůležitější.

### 1.3 Holografie

Holografií označujeme metodu záznamu optické informace, pomocí rekonstrukce optické vlnoplochy. Slovo holografie je složeninou dvou řeckých slov, které znamenají jednak holos - úplný a grafie - záznam, tedy úplný záznam. U konvenčních zobrazovacích technik, jako je fotografie, se zaznamenává pouze rozložení intenzity původní scény. V důsledku toho dojde ke ztrátě veškeré informace o fázi vln (optických drah do různých částí původní scény). Jedinečnou vlastností holografie je myšlenka nahrávání fáze i amplitudy světelných vln od objektu. Jelikož všechny záznamové materiály reagují pouze na intenzitu světla, je nutné převést informace o fázi do změny intenzity světla. Záznam interference fáze světelného signálu představuje tvar vlnoplochy, čili prostorovost zaznamenaného optického objektu. Holografie se skládá ze dvou kroků záznamu a následné rekonstrukce obrazu.

#### 1.3.1 Záznam hologramu

Hologram získáme tak, že optický svazek (obvykle používáme laserový paprsek kvůli jeho koherentním vlastnostem) se pomocí optického děliče rozdělí na dva svazky. První svazek, označovaný jako osvětlovací, necháme dopadat na předmět. Odrazem od tohoto předmětu získáme takzvaný objektový svazek. Tento svazek v sobě nese informace jak o rozložení intenzity světla na předmětu, tak o jeho prostorové struktuře, která je „ukryta“ ve fázi předmětové vlny.

Objektový svazek necháme dopadat na fotografickou desku, kde začne interferovat s druhým svazkem, takzvaným referenčním. Vzájemnou interferencí obou svazků budou vznikat na fotografické desce světlé a tmavé skvrny, kterým říkáme interferenční minima a maxima. Tyto „skvrny“ jsou od sebe vzdáleny srovnatelně s vlnovou délkou laseru, kterým ozařujeme daný objekt.

Matematické vyjádření komplexní amplituda objektové vlny je dána vztahem

$$E_o(x, y) = a_o(x, y) \exp(i \cdot \varphi_o(x, y)), \quad (14)$$

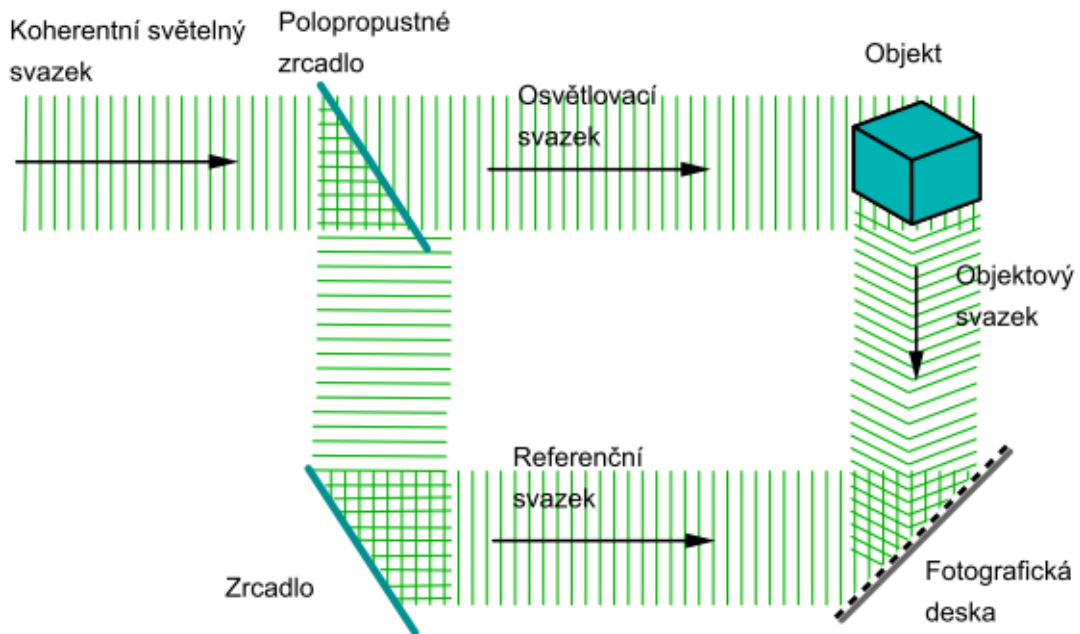
kde  $a_o$  je reálná amplituda a  $\varphi_o$  reálná fáze. Obdobně lze vyjádřit i komplexní amplituda referenční vlny podle vztahu

$$E_R(x, y) = a_R(x, y) \exp(i \cdot \varphi_R(x, y)), \quad (15)$$

kde  $a_R$  je reálná amplituda a  $\varphi_R$  reálná fáze.

Obě vlny interferují na povrchu záznamovém media. Výsledná intenzita je vypočítána podle vztahu

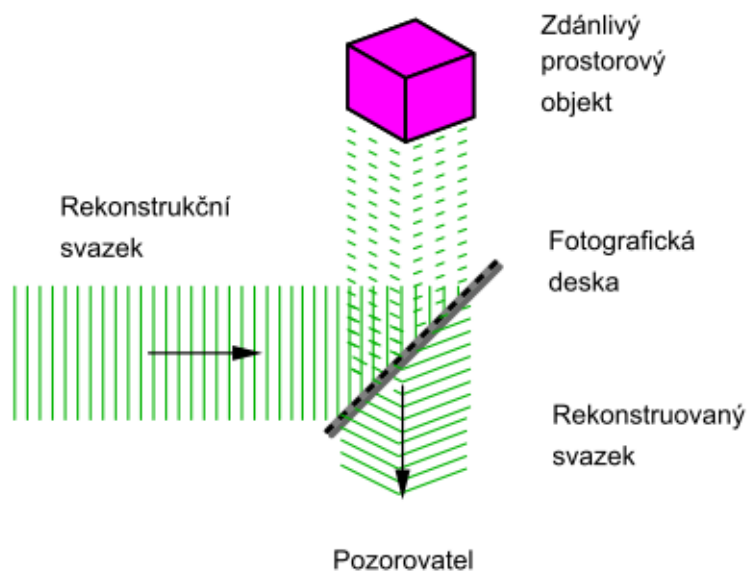
$$\begin{aligned} I(x, y) &= |E_O(x, y) + E_R(x, y)|^2 \\ &= (E_O(x, y) + E_R(x, y))(E_O(x, y) + E_R(x, y))^* \\ &= E_R(x, y) E_R^*(x, y) + E_O(x, y) E_O^*(x, y) + E_O(x, y) E_R^*(x, y) \\ &\quad + E_R(x, y) E_O^*(x, y). \end{aligned} \quad (16)$$



Obr 13. Záznam hologramu

### 1.3.2 Rekonstrukce hologramu

Proto, abychom mohli vidět informace uložené na hologramu, musíme provést takzvanou rekonstrukci hologramu. Rekonstrukce provedeme tak, že necháme na fotografickou desku dopadat rekonstrukční svazek pod stejným úhlem, jako dopadal původní referenční svazek. Díky difrakci rekonstrukčního svazku se vytvoří světelný svazek odpovídající trojrozměrnému obraz původního předmětu, takzvaný rekonstruovaný svazek.



Obr 14. Rekonstrukce Hologramu

### 1.3.3 Digitální holografie

Princip digitální holografie, je velmi podobný holografii klasické. Základní rozdíl mezi klasickou a digitální holografií spočívá ve způsobu záznamu hologramu. Zatím co u klasické holografie dopadá objektový a referenční svazek na fotocitlivou desku, u digitální holografie se záznam provádí pomocí CCD, případně CMOS čipů. Výsledný holografický záznam je následně elektronicky uložen a zpracován. Pro následnou rekonstrukci obrazu se používají různé numerické metody.

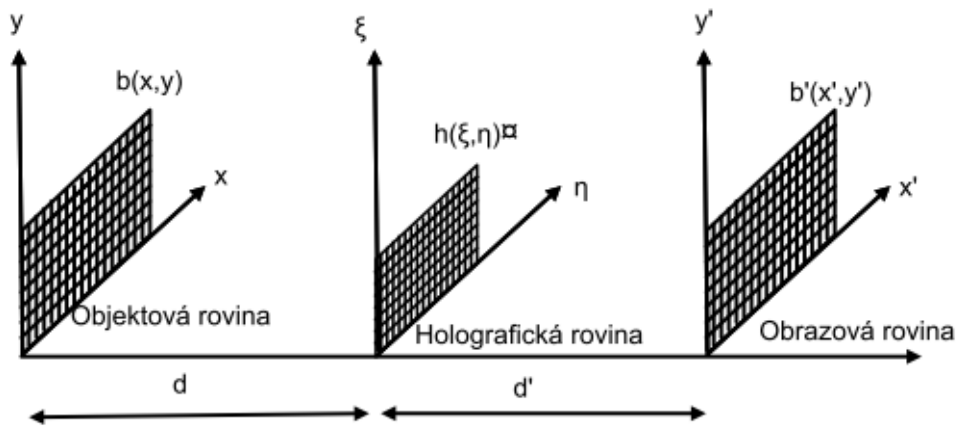
Prostorové rozlišení CCD-mřížky je mnohem menší než u holografické fólie, foto termoplastu nebo fotorefraktivního krystalu. Proto musíme dát pozor, aby byl dodržen tzv. Shannonův teorém. Ten říká, že frekvence vzorkovacího signálu musí být alespoň dvojnásobná oproti frekvenci měřeného signálu. V tomto případě tedy musí platit, že počet bodů CCD čipu, na délkovou jednotku, musí být alespoň dvojnásobkem počtu obrazových bodů. Z tohoto důvodu musí být holografické microinterferenční proužky

dostatečně vzdálené. Toho lze dosáhnout malou velikostí objektu, umístěním objektů daleko od CCD čipu nebo zmenšením objektové oblasti vhodně umístěnými čočkami. Referenční vlna musí dopadat pod velmi malým úhlem vůči objektové vlně.

Pro rekonstrukci digitálního hologramu se nepoužívá laserem emitovaný rekonstrukční svazek, ale nahraný a uložený hologram  $h(\xi, \eta)$  se vynásobí numerickým modelem, který odpovídá referenční vlně  $r(\xi, \eta)$ . V případě normálně dopadající rovinné vlny může být modelována jako  $r(\xi, \eta)=1$ . Pole v obrazové rovině  $b'(x', y')$  reprezentuje rekonstruovaný reálný obraz nahraného vlnového pole. To je spočítáno pomocí Rayleigh-Sommerfeldovou difrakční rovnice

$$b'(x', y') = \frac{1}{i\lambda} \iint h(\xi, \eta) r(\xi, \eta) \frac{e^{ik\rho}}{\rho} d\xi d\eta, \quad (17)$$

kde  $k=2\pi/\lambda$  odpovídá počtu vln,  $\lambda$  vlnové délce použitého laseru a  $\rho = \sqrt{d'^2 + (\xi - x')^2 + (\eta - y')^2}$ . V tomto případě je  $d'$  vzdálenost mezi CCD-mřížkou a obrazovou rovinou (viz Obr 15).



Diskrétní verze po Fresnelově aproximaci má tvar

$$b'(n, m) = A \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} h(k\Delta\xi, l\Delta\eta) r(k\Delta\xi, l\Delta\eta) e^{\frac{i\pi}{d'\lambda}(k^2\Delta\xi^2 + l^2\Delta\eta^2)} e^{2i\pi\left(\frac{kn}{N} + \frac{lm}{M}\right)}. \quad (18)$$

Pro uložený digitální hologram o velikosti  $N \times M$  a velikosti pixelů záznamového zařízení  $\Delta\xi \times \Delta\eta$ . Z tohoto vztahu můžeme jednoduše spočítat intenzitu pomocí vztahu



$$I(n, m) = |b'(n, m)|^2 \quad (19)$$

a fázi

$$\varphi(n, m) = \arctan \frac{\operatorname{Im}[b'(n, m)]}{\operatorname{Re}[b'(n, m)]}. \quad (20)$$

### 1.3.4 Holografická interferometrie

Holografická interferometrie je optickou zobrazovací metodu, založenou na interferometrickém posunu fází paprsků, odrážejících se od předmětu. Tato metoda umožňuje výzkum mechanických deformací povrchu, vibrací, malých posuvů objektu s přesností na zlomky velikosti vlnové délky použitého laseru.

Holografické interferometry mohou pracovat buď metodou dvojí expozice, nebo v reálném čase. Při práci metodou dvojí expozice se zaznamenávají dva různé stavy téhož transparentního objektu. Holografické interferometry pracující v reálném čase, zaznamenávají pouze jeden referenční stav transparentního objektu, ale při rekonstrukci interferuje rekonstruovaný předmětový svazek s reálným předmětovým svazkem procházejícím měřeným objektem v jiném stavu, a to v reálném čase.

Jednou z nejdůležitějších metod holografické interferometrie je takzvaná digitální holografická interferometrie. Tato metoda využívá vlastností digitální holografie a porovnává hodnoty dvou fází, které z hologramu získáme použitím rovnice (20). Výhodou této metody je fakt, že záznam i vyhodnocení jsou prováděny digitálně na počítači, což ji poskytuje široké možnosti dalšího využití.

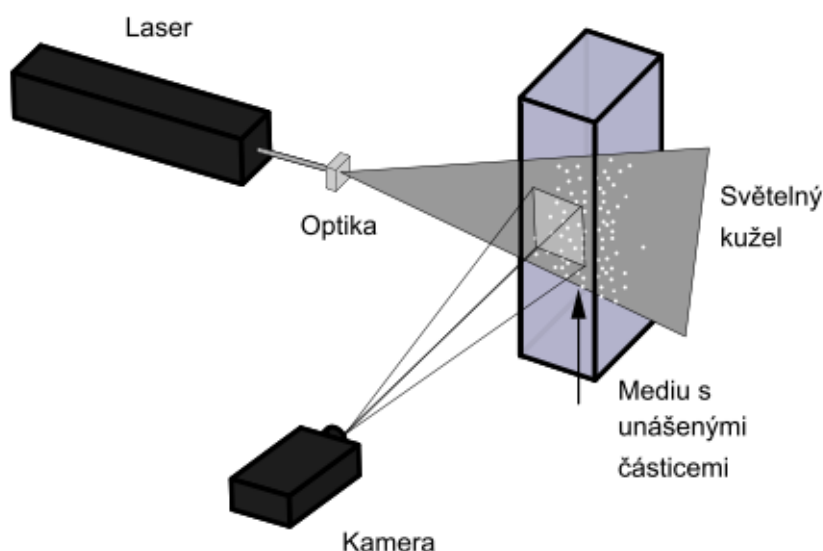
## 1.4 PIV

PIV je zkratka z anglických slov Particle Image Velocimetry, což se do češtiny překládá jako Integrovaná laserová anemometrie. Jde o techniku měření, která umožňuje získat informace o okamžitém rozložení rychlostí v dvourozměrném poli v proudící tekutině. Rychlosti se určují z měřené vzdálenosti, kterou urazí unášené částice v toku, a z času, který částice k překonání této vzdálenosti potřebují. Částice jsou buď součástí proudícího média, nebo se jimi toto médium sytí.

### 1.4.1 Princip PIV

Částice, obsaženy v proudící kapalině jsou osvětlovány rovinným světelným řezem. Tato světelná rovina je vytvářena vhodným výkonným laserem a systémem optických prvků. Abychom docílili stroboskopického jevu „zmrazení pohybu“ používá se pro buzení světelné roviny pulzní laser. Časový interval mezi jednotlivými pulzy je následně použit pro výpočet rychlosti částic v kapalině. Záznamový systém bývá tvořen většinou CCD kamerou, která je umístěna kolmo ke světelné rovině (viz Obr 16).

Kamera, zobrazuje částice, které jsou osvětleny světelnou rovinou, jako světlá zrna na tmavém podkladu. Pořizují se dva snímky velmi rychle za sebou, aby bylo možné určit rychlost částic ze změny polohy na obou snímcích.



Obr 16. Princip PIV

Celý snímek, pořízený CCD kamerou, je následně rozdělen na menší pravoúhlé plochy. Pro každou z těchto menších vyhodnocovacích ploch je korelační metodou určen jeden vektor, reprezentující průměrný posuv všech částic uvnitř daného prostoru. Následným dělením tohoto vektoru, podle vzorce

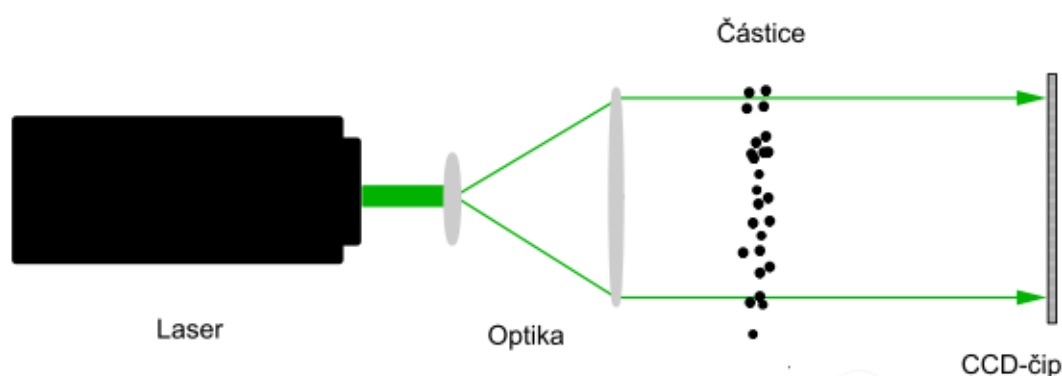
$$\vec{v} = \frac{\Delta \vec{X}}{\Delta t}, \quad (21)$$

kde  $\Delta \vec{X}$  je průměrný posuv částic a  $\Delta t$  je čas mezi dvěma záznamy, jsou vektory posunutí konvertovány do mapy nezpracovaných rychlostních vektorů. V další etapě se aplikují kontrolní algoritmy na nezpracované mapy a v nich detekují a odstraňují

extrémní hodnoty, jimiž jsou chybné vektory. Výstupem je nová vektorová mapa, ze které je možno další analýzou vytvářet proudnice, pole vířivosti atd.

## 1.5 HPIV

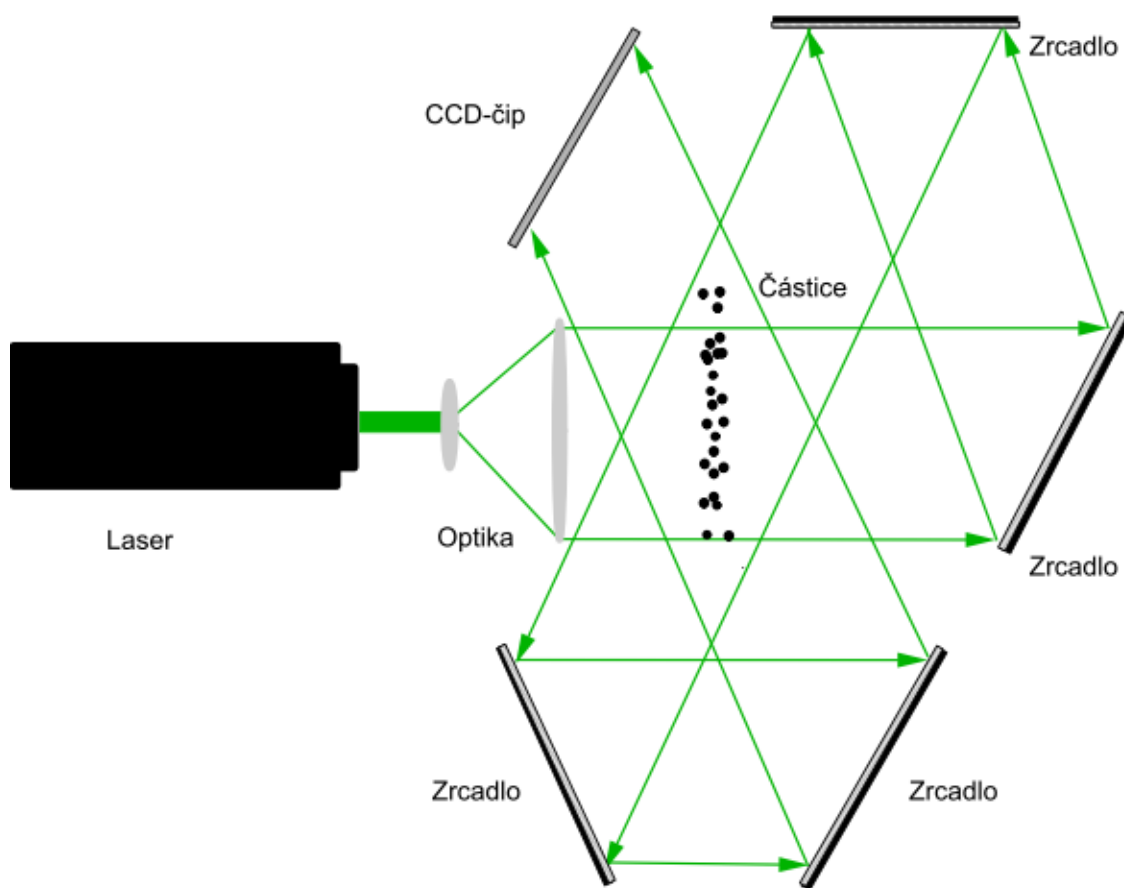
Holografická PIV (HPIV) je název měřicí metody, využívající vlastností digitální holografie pro zobrazení a zaznamenání částic v trojdimensionálním prostoru. Tato technika, se s výhodou využívá pro určení polohy, tvaru, velikosti nebo rychlosti částic v proudících kapalinách. Základní uspořádání je obdobné jako při použití klasické metody PIV s tím rozdílem, že zde musíme na záznamové zařízení přivést ještě referenční svazek.



Obr 17. Základní uspořádání in-line holografie

U této záznamové metody se často využívá takzvané in-line holografické uspořádání. U tohoto typu holografického uspořádání osvětluje laserový svazek drobné částice, část svazku se na částicích rozptýlí a dojde k difrakci, druhá část projde přímo bez difrakce a chová se tak jako referenční svazek. Tyto dva svazky následně interferují na CCD čipu, kde vzniká digitální hologram.

Na základním uspořádání (viz Obr 17), je průhledný objekt, který obsahuje malé částice. Objekt je osvětlen kolimovaným monochromatickým svazkem světla podél osy kolmé k CCD čipu. U in-line uspořádání provádíme rekonstrukci v řezech, které odpovídají jednotlivým vzdálenostem od záznamového zařízení. Následné vyhodnocení polohy z těchto řezů, můžeme získat pomocí porovnávání ostrosti jednotlivých řezů.



*Obr 18. Základní uspořádání tomografické holografie*

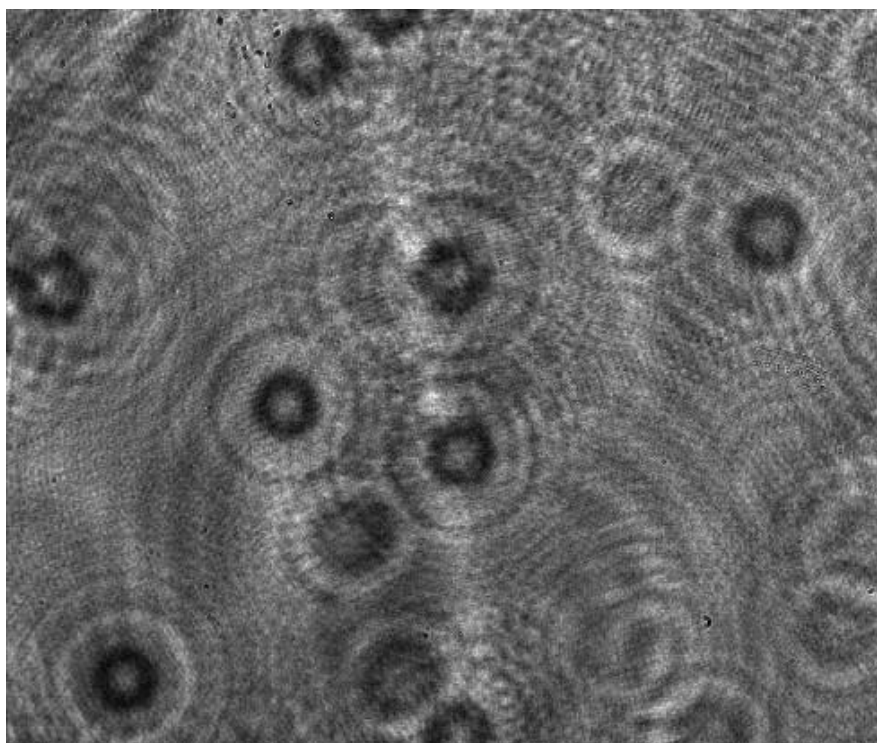
Dalším typem uspořádání, které se u HPIV využívá je tzv. tomografické uspořádání. U tohoto uspořádání (viz Obr 18), necháme kolimovaný laserový svazek, pomocí odrazů od čtyř zrcadel, projít celkem třikrát přes nádobu s drobnými částicemi, před dopadem na CCD čip. Pomocí digitální rekonstrukce hologramu, jsme schopni obraz zrekonstruovat v různých vzdálenostech od CCD čipu, pomocí čehož dostaneme tři komplexní pole odpovídající prvnímu, druhému nebo třetímu průchodu svazku přes částice. Protože optické svazky procházejí z různých směrů, jsme jejich zkombinováním schopni zrekonstruovat původní 3D objekt.

## 2 Experimentální část

### 2.1 Popis experimentu

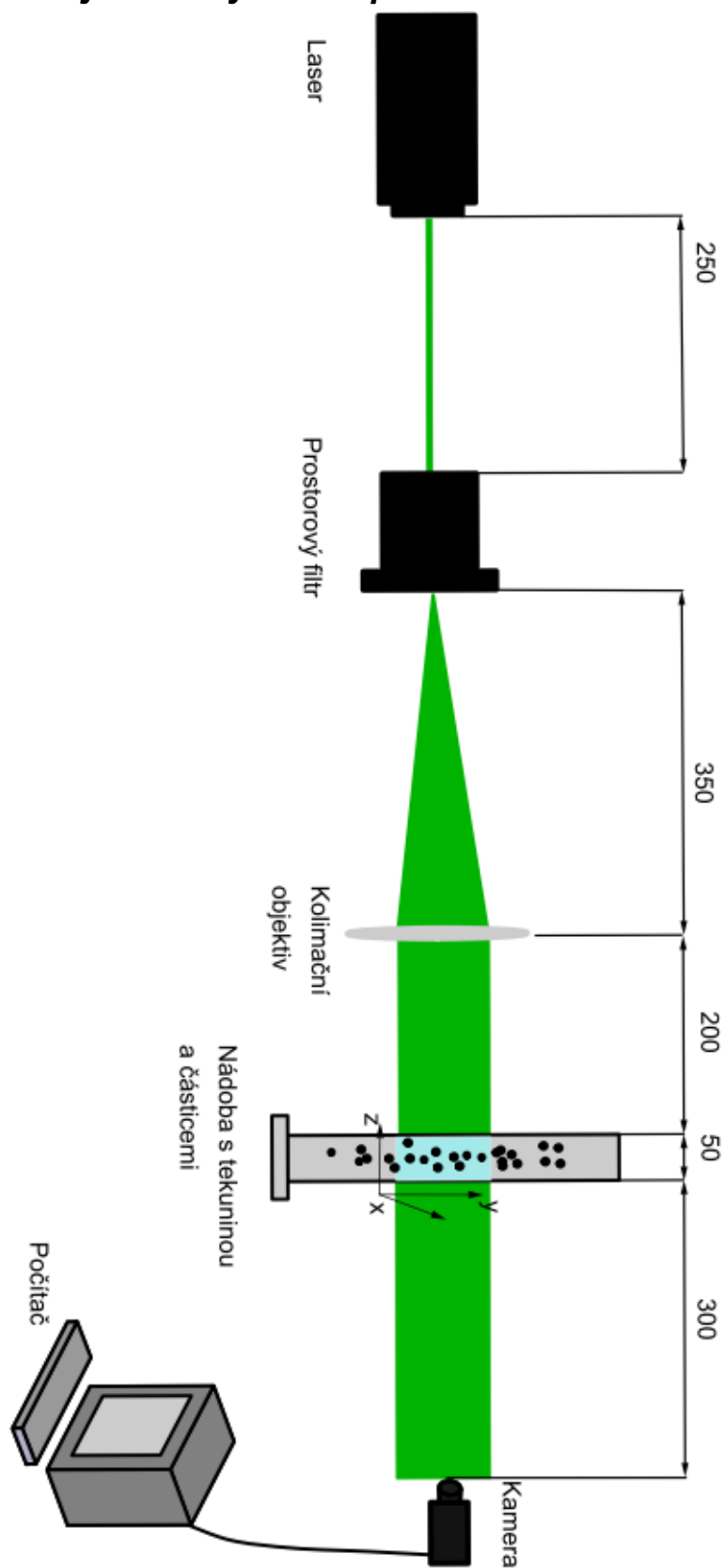
Při experimentu bylo použito rozvržení pro in-line holografii (viz Obr 20.), kde je osvětlovací svazek v jedné linii spolu s digitálním záznamovým zařízením. Jako částice, jejichž polohu jsme chtěli detekovat, byly použity 0,5 mm velké skleněné kuličky. Tyto kuličky byly umístěny v obdélníkové nádobě z čirého skla v roztoku 99%, glycerinového oleje. Pro záznam byla použita digitální kamera Stigray F-504.

Podle schématu (viz Obr 20.) jsme sestavili měřicí systém pro detekci polohy částic z digitálních hologramů. Laserový svazek jsme přivedli do seřízeného prostorového filtru, kde se odfiltrovaly vady svazku a optický svazek se roztáhl. Takto upravený optický svazek jsme zkolimovali vhodně nastaveným kolimačním objektivem. Výsledným svazkem jsme osvětlovali nádobu s kapalinou, do které jsme postupně přidávali skleněné kuličky. Díky tomuto uspořádání docházelo na CCD čipu digitální kamery, umístěné 30 cm za nádobou, ke vzniku hologramu, který tato kamera zaznamenala. Takto zaznamenaný hologram (viz. Obr 19.) jsme následě digitálně zpracovali pomocí navrženého softwaru v prostředí MATLAB.

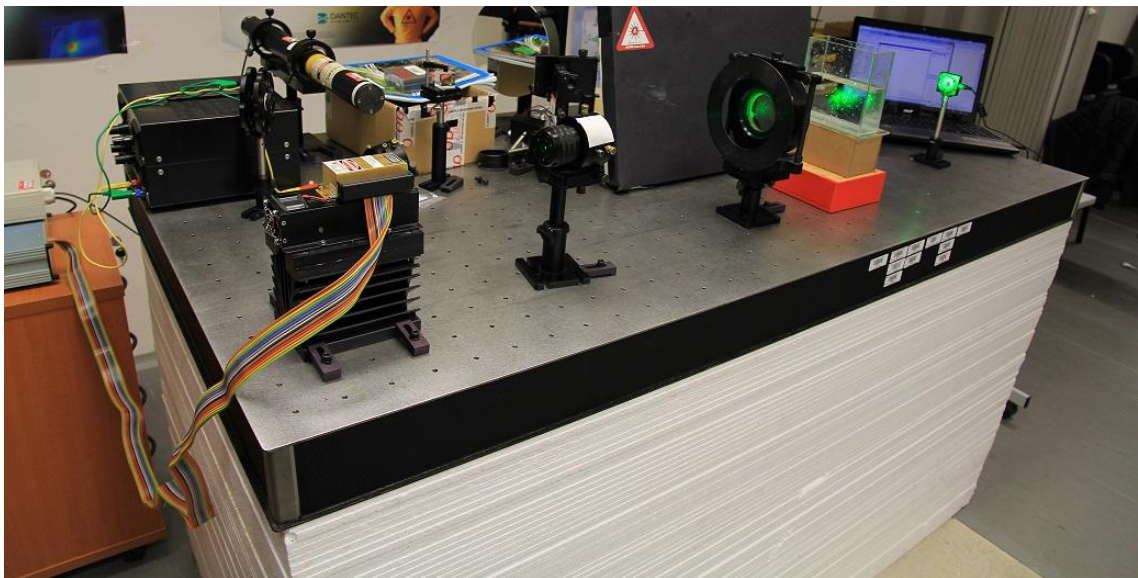


*Obr 19. Digitální hologram*

## 2.2 Rozvržení jednotlivých komponent



Obr 20. Schématické znázornění rozvržení jednotlivých komponent



*Obr 21. Reálné rozložení při experimentu*

## **2.3 Použité Komponenty**

### **Digitální Kamera - CCD**

Zkratka CCD pochází z anglického Charge-Coupled Device, což v překladu znamená zařízení s vázanými náboji. CCD využívá podobně jako všechny ostatní světlocitlivé součástky fotoelektrického jevu, kde se působením záření uvolňují elektrony, které unikají z látky. Po připojení elektrod nedochází k odvedení elektronů, jelikož elektroda je od polovodiče izolována tenoučkou vrstvičkou oxidu křemičitého.

Celá plocha senzoru se skládá z velkého počtu světlocitlivých pixelů, které se dopadem světla, pomocí výše zmíněného principu, nabíjí. Velikost náboje je úměrná intenzitě dopadajícího světla. Informace o intenzitě světla je v digitální formě předána k dalšímu zpracování. V případě kamery přímo připojené k počítači jsou informace odeslány na příslušný port a uloženy v počítači.



*Obr 22. AVT kamera Stingray F-504(převzato z [6])*



Samotný polovodičový prvek CCD je citlivý na intenzitu dopadajícího světla, nedokáže však odlišit jeho vlnovou délku. Chceme-li výsledný obraz barevný, musí být před každým pixelem CCD kamery umístěn barevný filtr (červený, modrý nebo zelený), který propustí jen vymezenou část spektra. Po zpracování informací o intenzitě jednotlivých barevně odlišených pixelů dojde ke sloučení obrazu a k reprodukci skutečné barvy dopadajícího světla.

Pro záznam našeho experimentu byla použita AVT kamera Stigray F-504. Tato kamera disponuje vysokým rozlišením 2452x2056 pixelů při rychlosti záznamu 9fps (9 snímků za sekundu). Relativně malá hodnota fps nám při této aplikaci nijak nevádí, neboť digitální kamera zde plní funkci spíše fotoaparátu než kamery.

### **Laser**

Jako monochromatické koherentní zdroje světla jsme použili Laser Compass 315M -150SL. Jedná se o zelený laser s vlnovou délkou  $\lambda=532$  nm s maximálním výstupním výkonem 150 mW. Polarizační poměr je u toho typu laseru 1:100.

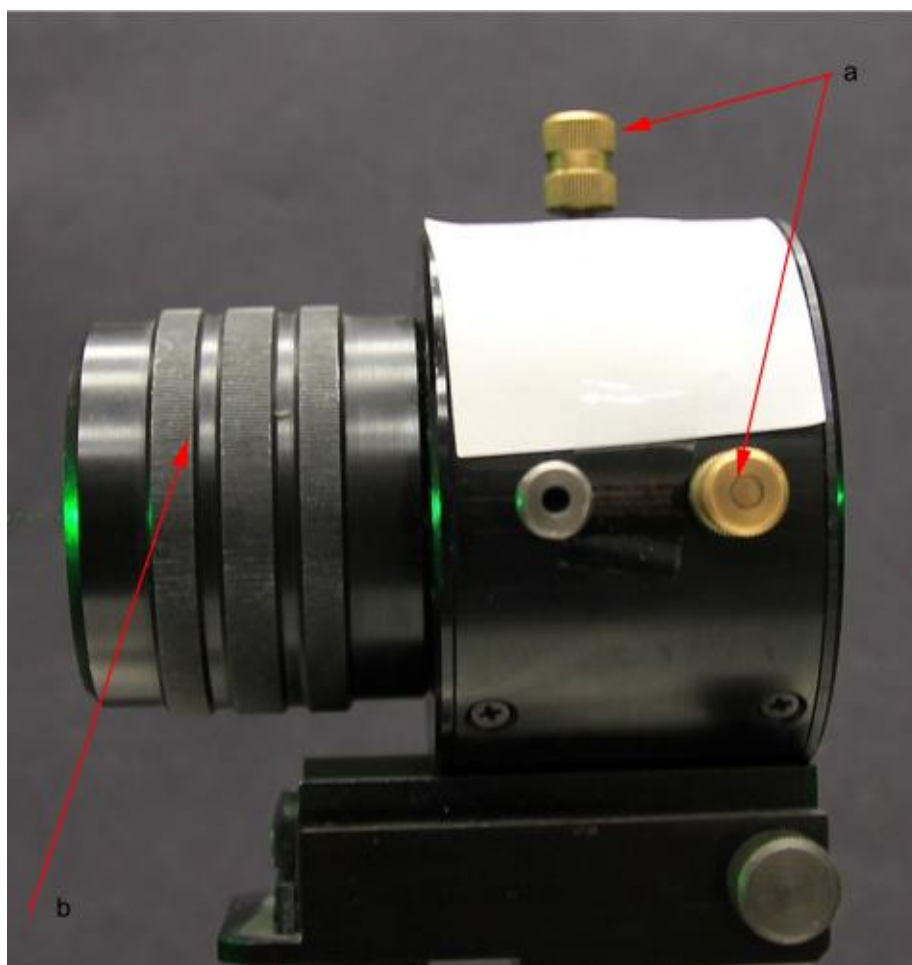


*Obr 23. Laser Compass 315M-150SL*

## Prostorová filtrace

Prostorová filtrace slouží k odstranění vad svazku, které zapříčiňují lokální změnu intenzity svazku. Tyto vady jsou zapříčiněny difrakčními a interferenčními jevy, které vznikají na pevných částicích, jako je prach, které ulpěly na povrchu optických prvků, bublinkách ve skle prvků, mechanických poruchách, apod.

Dopadá-li na optickou soustavu optický svazek s rovinnou vlnoplochou, dojde v blízkosti ohniskové roviny soustavy k vytvoření Fourierova ohybového obrazce zobrazeného předmětu. Zobrazení obrazu je možné jenom v tom případě, projdou-li ohniskovou rovinou alespoň dvě maxima, které spolu interferují. Pomocí dírková clony, která se vloží do ohniskové roviny tak, aby propouštěla pouze jedno maximum, zamezíme vzniku obrazu.



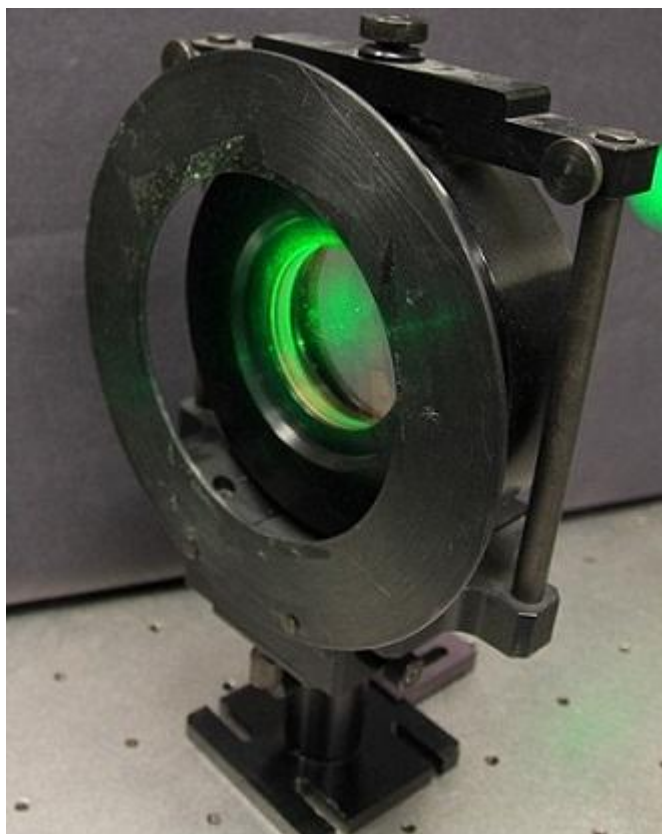
Obr 24. Prostorový filtr, a) mikrometrické šrouby pro posuv clonky v horizontálním a vertikálním směru, b) mikroposuv objektivu ve směru clonky

Základními prvky prostorového filtru (viz. Obr 24) jsou tedy mikroskopový objektiv a dírková clona, která má možnost pohybu ve dvou, na sebe kolmých směrech. Pro nastavení prostorového filtru je nejprve použita clonka s větším otvorem (cca 5mm), která je po ustavení objektivu s clonkou do osy svazku nahrazena clonkou s vhodnou velikostí otvoru. Objektiv je nastaven tak, aby clonka byla ve vzdálenosti větší, než je jeho ohnisková vzdálenost. Posuvem objektivu k clonce je možné pozorovat na stínítku tzv. Interferenční proužky. Prostorový filtr je seřízen, když vymizí soustava interferenčních proužků kolem stopy a na stínítku je vidět čistý svazek bez parazitní interference.

Velikosti otvoru clonek se pohybují v řádu desítek  $\mu\text{m}$  a jejich nejvhodnější kombinace s určitým mikroobjektivem je dána víceméně experimentálně. Velikost otvoru námi použitého prostorového filtru byla 10  $\mu\text{m}$ .

### **Kolimace**

Nejvhodnějším způsobem zpracování optických svazků v holografickém uspořádání je způsob, kdy je svazek rozveden po celé sestavě v původní formě laserového paprsku a teprve v poslední fázi se upravuje.

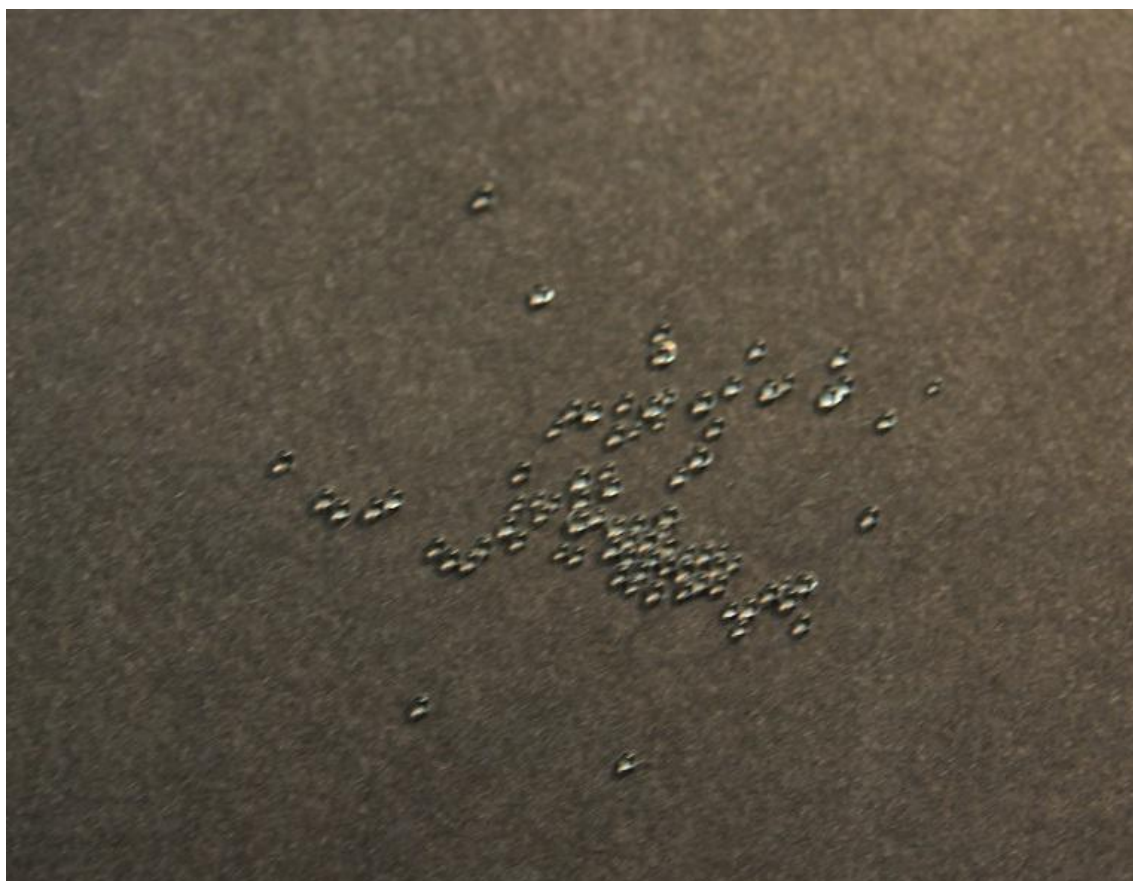


*Obr 25. Kolimační objektiv*

Kolimovaným svazkem v geometrické optice rozumíme takový svazek, který má rovnoběžné paprsky. Pro gaussovský svazek není možné takového stavu dosáhnout, protože tento svazek má vždy v podélném průřezu kaustickou křivku ve tvaru hyperboly. Přesto však díky kolimačnímu objektivu (viz. Obr 25.) získáme svazek víceméně kolimovaný, alespoň v tom smyslu, že má daleko větší průřez než vstupní svazek a velmi malou rozbíhavost. Pro účely našeho experimentu byl použit objektiv s ohniskovou vzdáleností 200m.

### Částice

Jako vhodné částice k detekci byly v experimentu použity skleněné kuličky o průměru 0,5 mm.



*Obr 26. Částice 0,5 mm*

## **Optický stůl**

Optický stůl je plocha uspůsobená pro rozmísťování a uchycování optických prvků. Uchycení je realizováno pomocí pomocných držáčků a šroubků. Po celé ploše stolu jsou rozmístěny v pravidelných intervalech závitové uchycení držáčku. Jelikož většina optických měření je náchylná na mechanické chvění, musí být optický stůl dobře odtlumen. V našem případě bylo toto odtlumení realizováno polystyrenovými deskami, naskládanými mezi deskou stolu a podlahou laboratoře.

## **2.4 Softwarová implementace**

Software pro vyhodnocení rozložení částic v kapalině je implementován v prostředí skriptovacího jazyka MATLAB. Prostředí MATLAB (z MATrix LABoratory) je integrovaný systém pro vědeckotechnické výpočty, modelování, návrhy algoritmů, simulace, analýzu a prezentaci dat, paralelní výpočty, měření a zpracování signálů, návrhy řídicích a komunikačních systémů. MATLAB je nástroj pro vývoj širokého spektra aplikací, umožňuje poměrně efektivně implementovat i náročné matematické vzorce a poskytuje velké množství nástrojů k manipulaci s nimi. Další výhodou tohoto prostředí je možnost využití již dříve navržených funkcí pro rekonstrukci digitálního hologramu, které jsou také implementovány v tomto prostředí. A proto je využití skriptovacího jazyka MATLAB maximálně efektivní.

Program pracuje se zaznamenanými digitálními hologramy, které nejprve rekonstruuje a následně vyhodnotí a vybere místa výskytu, předem specifikovaných, částic. Takto získané údaje následně použije pro vyznačení a vykreslení těchto částic do 3D modelu.

### **2.4.1 Princip softwaru**

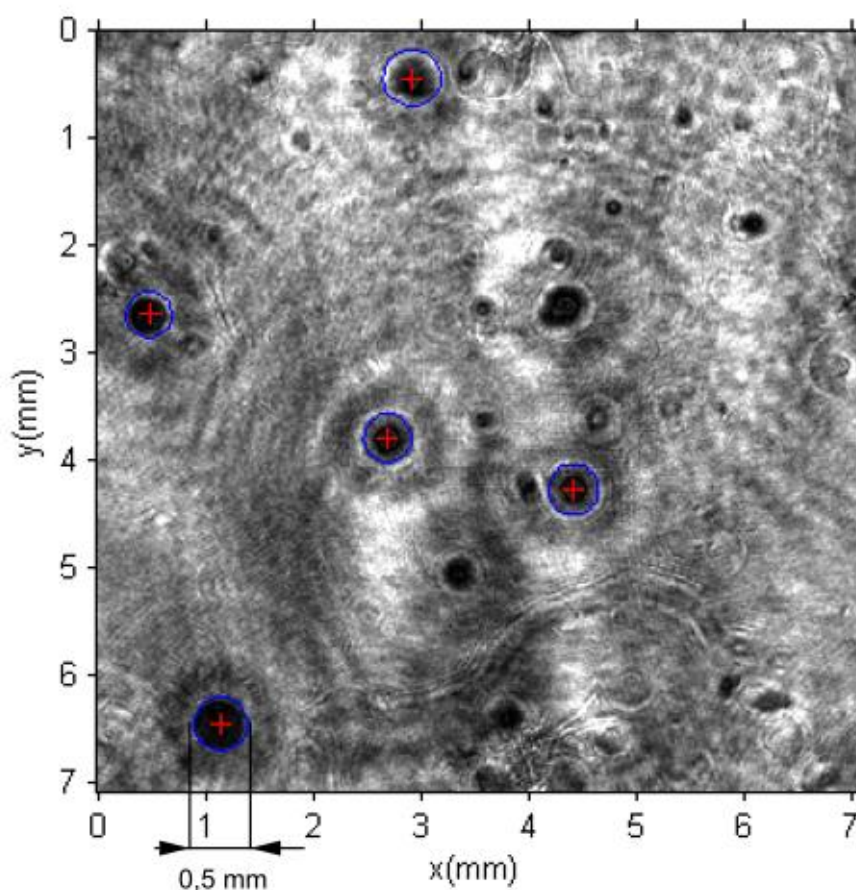
Program nejprve pomocí funkce `conv_rec(h,lambda,d,pixx,pixy,Ur,1)`, která rekonstruuje komplexní vlnové pole z digitálního hologramu (viz. Obr 19.), provede rekonstrukci původního digitálního hologramu. Funkce `conv_rec` používá pro svoji funkci konvoluční metodu rekonstrukce, která má oproti Fresnelovské metodě tu výhodu, že všechny rekonstruované obrazy mají stejnou velikost bez ohledu na vzdálenost od objektu, v jaké rekonstrukci provádíme. Funkce `conv_rec(h,lambda,d,pixx,pixy,Ur,1)` pomocí vzorce (18) vrací jediný parametr, a sice



b, což je 2D komplexní pole. Z něj můžeme použitím vzorců (19) respektive (20), získat intenzitu a fázi.

Funkce `conv_rec` ke svému vyhodnocení využívá tyto parametry:  $h$  - pole komplexního hologramu,  $\lambda$  - vlnová délka použitého laseru [m],  $d$  - vzdálenost CCD a měřeného objektu,  $\text{pixx}$  resp.  $\text{pixy}$  - velikost pixelu použitého detektoru [m],  $U_r$  - komplexní amplituda rekonstrukční vlny (defaultně  $U_r=1$ ),  $M$  - scaling factor nastavuje zvětšení (zmenšení) rekonstruovaného pole (pro "zmenšení" objektu  $M < 1$ , defaultně  $M$  odpovídá difrakčnímu limitu!),  $s_k$  resp.  $s_l$  - posun objektu v horizontálním resp. vertikálním směru (defaultně  $s_k = s_l = 1$ ).

Tento postup se opakuje čtyřikrát v námi určeném rozmezí. Výsledkem této operace jsou čtyři zrekonstruované obrazy ve čtyřech různých vzdálenostech od záznamového zařízení, které jsme si předtím určily.



Obr 27. Identifikace částic

Na těchto obrazech se pomocí funkce `CircularHough_Grd(img, radrange)` (viz. příloha) určí pozici a velikost námi specifikovaných kruhových objektů (viz. Obr

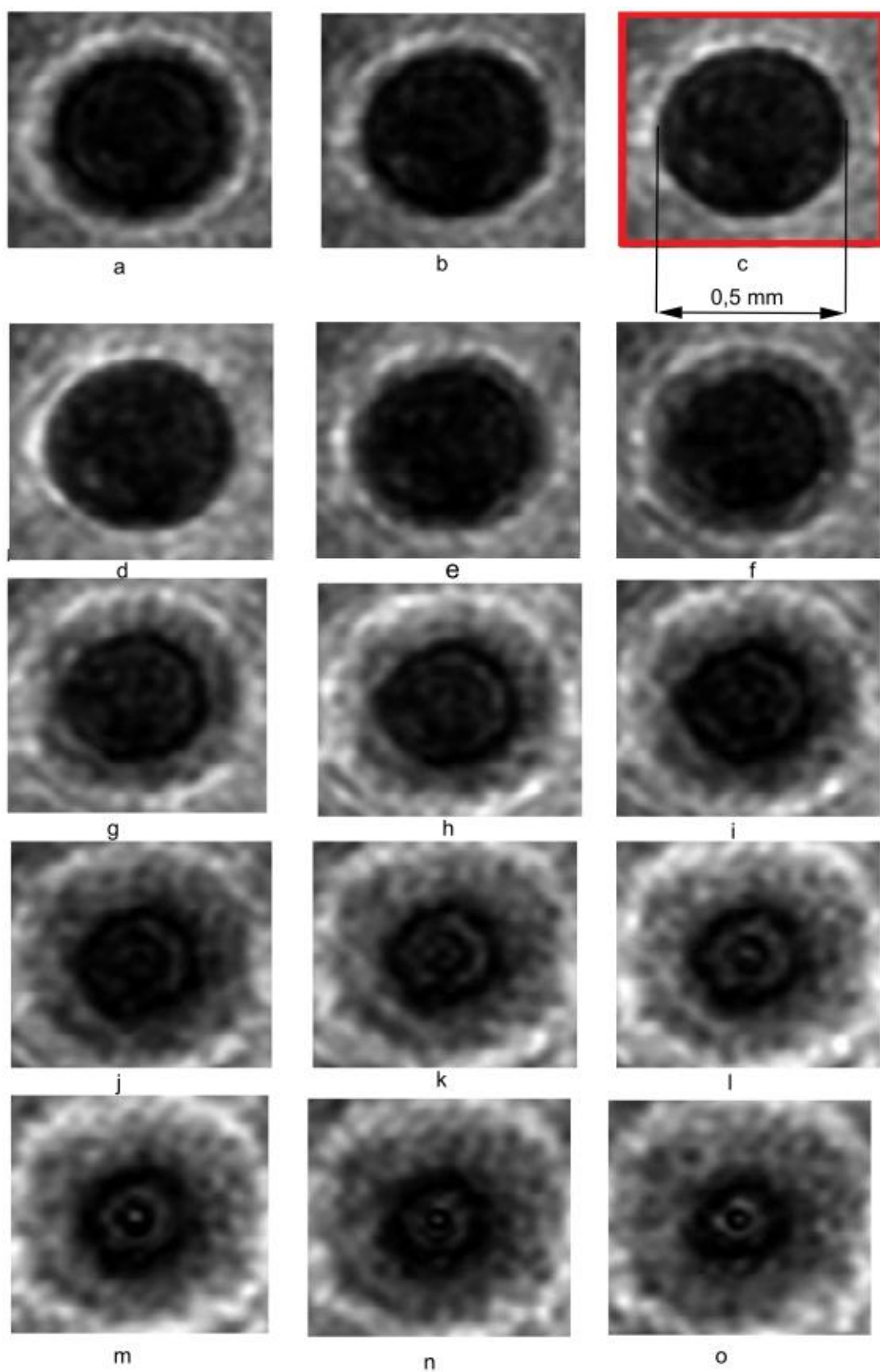
27). K tomu tato funkce využívá Hougovy transformace, metody pro nalezení parametrického popisu objektů v obraze na základě gradientu oblastí. Funkce CircularHough\_Grd pro svoji funkci vyžaduje tyto parametry: img - obraz v němž jsou kruhové tvary detekovány, radrange – minimální a maximální velikost detekovaných kruhů. Výstupem jsou pak dvě pole, první velikostí odpovídající počtu detekovaných částic obsahující pozice středů a druhé, stejně velké odpovídající jejich průměrům.

Z důvodů zabránění detekce nežádoucích předmětů (vzduchové bublinky, nečistoty), bylo nutné stanovit podmínku velikosti detekovaných částic. Takto stanovená podmínka je vhodným kompromisem mezi dostatečnou přesností při rozpoznávání částic a výpočetním časem, který by bez této podmínky byl neúnosně velký.

Po rozpoznání částic a určení jejich polohy na zrekonstruovaném hologramu, tedy jejich souřadnice ve 2D prostoru, je potřeba určit jejich vzdálenost od záznamového zařízení. Tato vzdálenost nám dává informaci o třetím rozměru a spolu s předešlou detekcí polohy, úplnou informaci o poloze částic v prostoru. K tomu se v programu využívá porovnávání gradientů jednotlivých výřezů okolí středů detekovaných částic. Porovnáním gradientů (viz Obr 28) určíme výřez, který je nejostřejší, a tedy ten, jenž odpovídá vzdálenosti, ve které se nachází detekovaná částice.

Postupně provádíme rekonstrukci původního digitálního hologramu v celém rozsahu námi požadované oblasti. Přičemž vždy vybereme oblast okolo předem detekovaných bodů a spočítáme jejich gradient. Je potřeba volit dostatečně jemný krok, abychom docílili co nejlepší přesnosti. Následně porovnáme velikosti gradientů jednotlivých výřezů. Výřez, který má největší gradient je ten nejostřejší a tedy ten, který odpovídá vzdálenosti částice od záznamového zařízení.

Po detekci a zaostření všech částic program všechny získaná data vypíše a vykreslí do 3D modelu (viz. Obr 29), kde souřadnice X a Y odpovídají pozici na zrekonstruovaném hologramu a Z je vzdálenost od počátku detekční oblasti nastavené při zaostřování v programu.

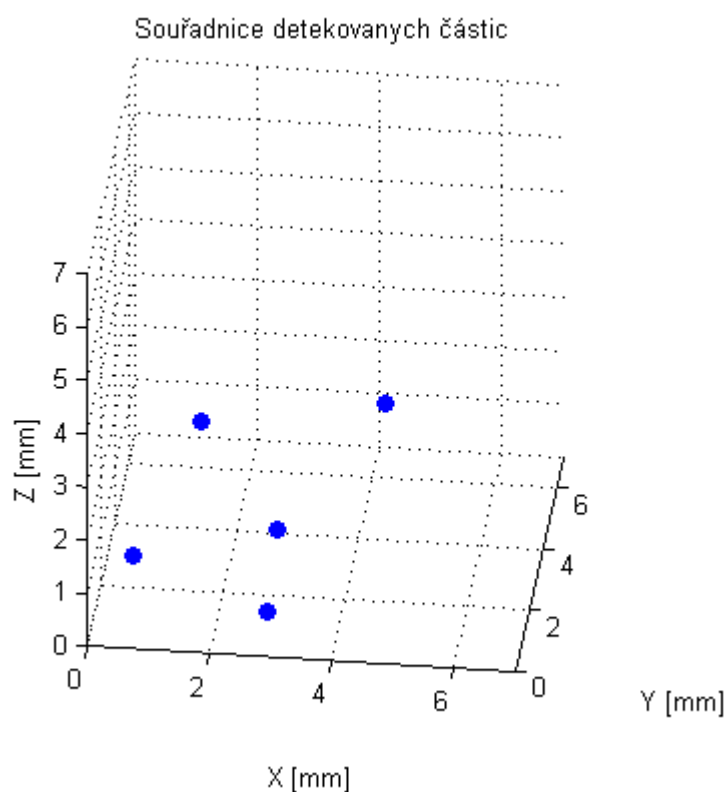


Obr 28. Postupné zaostřování detekované částice v postupných vzdálenostech od CCD,  $a=30,02$  cm,  $b=30,04$  cm,  $c=30,06$  cm,  $d=30,08$  cm,  $e=31$  cm,  $f=31,02$  cm,  $g=31,04$  cm,  $h=31,06$  cm,  $i=31,08$  cm,  $j=32$  cm,  $k=32,02$  cm,  $l=32,04$  cm,  $m=32,06$  cm,  $n=32,08$  cm,  $o=33$  cm . - Označený obrázek byl zvolen jako nejostřejší



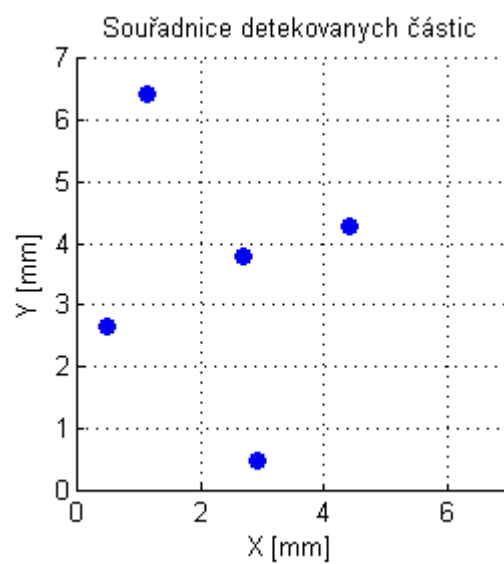
### 3 Výsledky experimentu

Výsledkem experimentální části je detekce polohy pěti kuliček a jejich následné znázornění do 3D modelu (viz. Obr 29.). Detekovaná poloha částic odpovídá skutečnému rozložení kuliček v nádobě v době zaznamenání hologramu. Tato informace spolu s faktem, že rozměry detekovaných částic odpovídají rozměrů našich kuliček pouze s drobnou odchylkou (viz. tab 1.), je důkazem, že se nám skutečně podařilo rozpoznat tyto částice, za pomoci digitální holografie v kapalině.

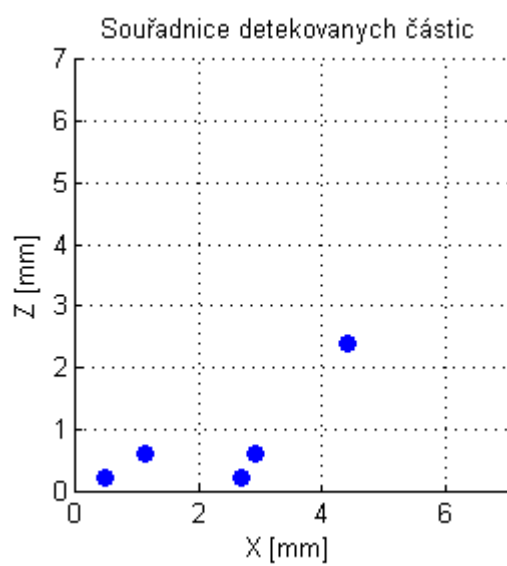


Obr 29. Zobrazení pozice částic ve 3D mapě

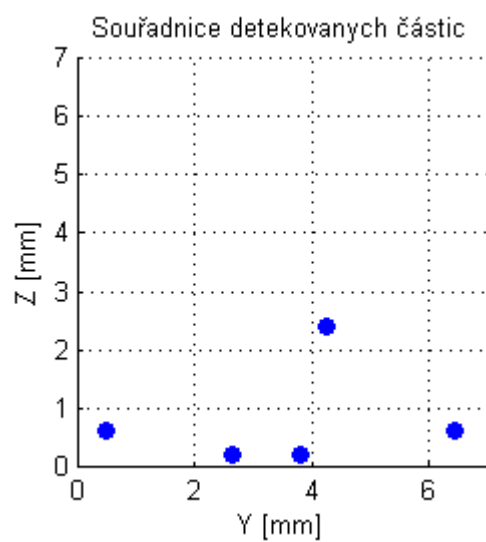
Pro lepší představu o rozložení kuliček v prostoru, jsou zde rovněž zobrazeny pozice jednotlivých kuliček z pohledů os X-Y (viz. Obr 30.), X-Z (viz. Obr 31.) a Y-Z (viz. Obr 32.). Z důvodu větší přehlednosti je nula pro Z-tovou souřadnici na začátku rozsahu detekované oblasti, v našem případě ve vzdálenosti 30 cm od CCD čipu.



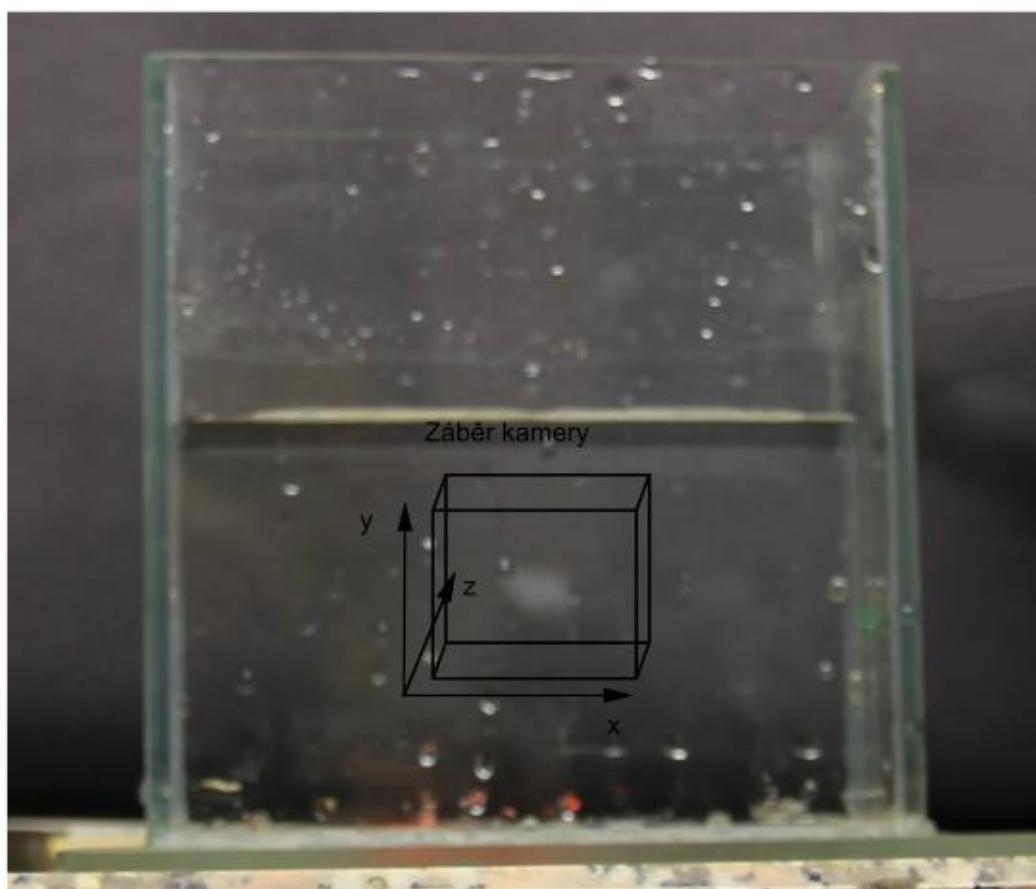
Obr 30. Zobrazení pozice částic v rovině X-Y



Obr 31. Zobrazení pozice částic v rovině X-Z



Obr 32. Zobrazení pozice částic v rovině Y-Z



Obr 33. Reálné rozmístění částic v nádobě

tab 1. Tabulka souřadnic a průměrů detekovaných částic

Částice	Souřadnice			Průměr [mm]
	X[mm]	Y[mm]	Z[mm]	
1.	0,466	2,651	0,200	0,435
2.	2,679	3,795	0,200	0,442
3.	2,904	0,457	0,600	0,531
4.	4,397	4,270	2,400	0,469
5.	1,132	6,445	0,600	0,490

## 4 Závěr

Cílem bakalářské práce byla teoretická studie základů holografie, problematiky PIV a holografického PIV. Na základě vypracované rešerše navrhnout a sestavit holografické uspořádání pro detekci polohy částic. Na závěr bylo nutné naprogramovat software pro vyhodnocení a grafické zobrazení údajů získaných experimenty.

Klasické metody PIV, které jsou dnes hojně využívány, jsou poměrně dobrým zdrojem informací, pokud nás zajímá pouze detekce nebo změna polohy částic ve dvoudimensionálním prostoru. Neposkytují však informace o chování částic a potažmo celé kapaliny v jejím celém objemu. Pro tyto účely je tedy nezbytné využít jiných prostředků. Jedním z těchto prostředků je právě holografická PIV, která nám tyto informace je schopna poskytnout. Bakalářská práce popisuje základní využití digitální holografie pro detekci částic a její výsledky potvrzují teoretické předpoklady, neboť se podařilo detekovat polohu i velikost částic v celém měřeném objemu.

Jako první byla provedena základní rešerše nejdůležitějších pojmů a principů, které byly pro splnění bakalářské práce nezbytné.

V druhé, experimentální, části jsme s využitím poznatků z rešerše navrhli a sestavili potřebné holografické uspořádání. Nejvýhodnějším uspořádáním z hlediska hardwarové náročnosti i zpracování získaných dat se ukázalo in-line holografické uspořádání, kdy jsou referenční i objektová vlna paralelní. Při experimentu se podařilo zaznamenat velké množství hologramů pro různé částice v různých druzích tekutin.

Softwarová implementace probíhala v prostředí MATLAB, které je pro implementaci složitých matematických algoritmů ideální. Kromě toho, software pro rekonstrukci digitálních hologramů, který byl využíván, je rovněž naprogramován v prostředí MATLAB a proto bylo jeho využití programu maximálně efektivní. Výsledný software je univerzální a po zadání počátečních parametrů uživatelem do inicializační části zdrojového kódu detekuje částice v měřeném objemu zcela automaticky.

Pro další možnosti využívání softwaru v budoucnosti je doporučeno doplnit software o uživatelské rozhraní, které nastavování požadovaných parametrů zpřehlední a zjednoduší. Dalším možným vylepšením softwaru by mohla být časová optimalizace výpočetního procesu, protože pro větší množství vyhodnocovaných dat může celková časová náročnost dosahovat i desítek minut.

Metoda měření i software pro zpracování dat budou v budoucnu rozšířeny o možnost měření rychlosti částic v tekutině.

V příloze je v MATLABu naprogramovaný software pro detekci polohy částic, který je rovněž na přiloženém CD.

## Seznam použité literatury

- [1] KREIS, T.: *Handbook of holographic Interferometry: Optical and Digital Methods*, Berlin: Wiley, 2004, p. 542.
- [2] SALEH, B. E. A.; TEICH, M.: *Základy fotoniky 1. díl*, Praha: Matfyzpress, 1994.
- [3] SALEH, B. E. A.; TEICH, M.: *Základy fotoniky 2. díl*, Praha: Matfyzpress, 1994.
- [4] SALEH, B. E. A.; TEICH, M.: *Základy fotoniky 3. díl*, Praha: Matfyzpress, 1995.
- [5] J. FUKA a B. HAVELKA, *Optika*, Praha: SPN, 1961.
- [6] Alliedvisiontec [Online]. [cit 10.11.2011]. Dostupné na www: <<http://www.alliedvisiontec.com/emea/products/cameras/firewire/stingray/f-504bc.html>>.
- [7] J. NUSSBERGER, *Fyzika Optika*, Praha: Státní nakladatelství technické literatury Praha, 1962.
- [8] R. P. FEYNMAN, R. B. LEIGHTON a M. SANDS, *Feynmanovy přednášky z fyziky 1/3*, Praha: Fragment, 2010.
- [9] MathWorks : MATLAB and Simulink for Technical Computing [online]. [cit. 12.10.2011]. Dostupné na www: <<http://www.mathworks.com/>>.

## Příloha

### Příloha 1 – Program CircularHough\_Grd

```
function [accum, varargout] = CircularHough_Grd(img, radrange,
varargin)
%Detect circular shapes in a grayscale image. Resolve their center
%positions and radii.
%
% [accum, circen, cirrad, dbg_LMmask] = CircularHough_Grd(
%     img, radrange, grdthres, fltr4LM_R, multirad, fltr4accum)
% Circular Hough transform based on the gradient field of an image.
% NOTE: Operates on grayscale images, NOT B/W bitmaps.
%       NO loops in the implementation of Circular Hough
transform,
%               which means faster operation but at the same time
larger
%               memory consumption.
%
% INPUT: (img, radrange, grdthres, fltr4LM_R, multirad,
fltr4accum)
%
% img:         A 2-D grayscale image (NO B/W bitmap)
%
% radrange:    The possible minimum and maximum radii of the circles
%               to be searched, in the format of
%               [minimum_radius , maximum_radius] (unit: pixels)
%               **NOTE**: A smaller range saves computational time
and
%               memory.
%
% grdthres:    (Optional, default is 10, must be non-negative)
%               The algorithm is based on the gradient field of the
%               input image. A thresholding on the gradient magnitude
%               is performed before the voting process of the Circular
%               Hough transform to remove the 'uniform intensity'
%               (sort-of) image background from the voting process.
%               In other words, pixels with gradient magnitudes
smaller
%               than 'grdthres' are NOT considered in the computation.
%               **NOTE**: The default parameter value is chosen for
%               images with a maximum intensity close to 255. For
cases
%               with dramatically different maximum intensities, e.g.
%               10-bit bitmaps in stead of the assumed 8-bit, the
default
%               value can NOT be used. A value of 4% to 10% of the
maximum
%               intensity may work for general cases.
%
% fltr4LM_R:   (Optional, default is 8, minimum is 3)
%               The radius of the filter used in the search of local
%               maxima in the accumulation array. To detect circles
whose
%               shapes are less perfect, the radius of the filter
needs
%               to be set larger.
%
% multirad:    (Optional, default is 0.5)
%               In case of concentric circles, multiple radii may be
```





```

        error('CircularHough_Grd: ''img'' has to be larger than 32-by-32');
    end

    if numel(radrange) ~= 2 || ~isnumeric(radrange),
        error(['CircularHough_Grd: ''radrange'' has to be ', ...
            'a two-element vector']);
    end

    prm_r_range = sort(max( [0,0;radrange(1),radrange(2)] ));

    % Parameters (default values)
    prm_grdthres = 10;
    prm_fltrLM_R = 8;
    prm_multirad = 0.5;
    func_compu_cen = true;
    func_compu_radii = true;

    % Validation of arguments
    vap_grdthres = 1;
    if nargin > (1 + vap_grdthres),
        if isnumeric(varargin{vap_grdthres}) && ...
            varargin{vap_grdthres}(1) >= 0,
            prm_grdthres = varargin{vap_grdthres}(1);
        else
            error(['CircularHough_Grd: ''grdthres'' has to be ', ...
                'a non-negative number']);
        end
    end

    vap_fltr4LM = 2; % filter for the search of local maxima
    if nargin > (1 + vap_fltr4LM),
        if isnumeric(varargin{vap_fltr4LM}) && varargin{vap_fltr4LM}(1) >=
3,
            prm_fltrLM_R = varargin{vap_fltr4LM}(1);
        else
            error(['CircularHough_Grd: ''fltr4LM_R'' has to be ', ...
                'larger than or equal to 3']);
        end
    end

    vap_multirad = 3;
    if nargin > (1 + vap_multirad),
        if isnumeric(varargin{vap_multirad}) && ...
            varargin{vap_multirad}(1) >= 0.1 && ...
            varargin{vap_multirad}(1) <= 1,
            prm_multirad = varargin{vap_multirad}(1);
        else
            error(['CircularHough_Grd: ''multirad'' has to be ', ...
                'within the range [0.1, 1]']);
        end
    end

    vap_fltr4accum = 4; % filter for smoothing the accumulation array
    if nargin > (1 + vap_fltr4accum),
        if isnumeric(varargin{vap_fltr4accum}) && ...
            ndims(varargin{vap_fltr4accum}) == 2 && ...
            all(size(varargin{vap_fltr4accum}) >= 3),
            fltr4accum = varargin{vap_fltr4accum};
        else
            error(['CircularHough_Grd: ''fltr4accum'' has to be ', ...

```

```

        'a 2-D matrix with a minimum size of 3-by-3']);
    end
else
    % Default filter (5-by-5)
    fltr4accum = ones(5,5);
    fltr4accum(2:4,2:4) = 2;
    fltr4accum(3,3) = 6;
end

func_compu_cen = ( nargout > 1 );
func_compu_radii = ( nargout > 2 );

% Reserved parameters
dbg_on = false;          % debug information
dbg_bfigno = 4;
if nargout > 3,  dbg_on = true;  end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Convert the image to single if it is not of
% class float (single or double)
img_is_double = isa(img, 'double');
if ~(img_is_double || isa(img, 'single')),
    imgf = single(img);
end

% Compute the gradient and the magnitude of gradient
if img_is_double,
    [grdx, grdy] = gradient(img);
else
    [grdx, grdy] = gradient(imgf);
end
grdmag = sqrt(grdx.^2 + grdy.^2);

% Get the linear indices, as well as the subscripts, of the pixels
% whose gradient magnitudes are larger than the given threshold
grdmasklin = find(grdmag > prm_grdthres);
[grdmask_IdxI, grdmask_IdxJ] = ind2sub(size(grdmag), grdmasklin);

% Compute the linear indices (as well as the subscripts) of
% all the votings to the accumulation array.
% The Matlab function 'accumarray' accepts only double variable,
% so all indices are forced into double at this point.
% A row in matrix 'lin2accum_aJ' contains the J indices (into the
% accumulation array) of all the votings that are introduced by a
% same pixel in the image. Similarly with matrix 'lin2accum_aI'.
rr_4linaccum = double( prm_r_range );
linaccum_dr = [ (-rr_4linaccum(2) + 0.5) : -rr_4linaccum(1) , ...
    (rr_4linaccum(1) + 0.5) : rr_4linaccum(2) ];

lin2accum_aJ = floor( ...
    double(grdx(grdmasklin)./grdmag(grdmasklin)) * linaccum_dr + ...
    repmat( double(grdmask_IdxJ)+0.5 , [1,length(linaccum_dr)] ) ...
);
lin2accum_aI = floor( ...
    double(grdy(grdmasklin)./grdmag(grdmasklin)) * linaccum_dr + ...
    repmat( double(grdmask_IdxI)+0.5 , [1,length(linaccum_dr)] ) ...
);

```

```

% Clip the votings that are out of the accumulation array
mask_valid_aJaI = ...
    lin2accum_aJ > 0 & lin2accum_aJ < (size(grdmag,2) + 1) & ...
    lin2accum_aI > 0 & lin2accum_aI < (size(grdmag,1) + 1);

mask_valid_aJaI_reverse = ~ mask_valid_aJaI;
lin2accum_aJ = lin2accum_aJ .* mask_valid_aJaI +
mask_valid_aJaI_reverse;
lin2accum_aI = lin2accum_aI .* mask_valid_aJaI +
mask_valid_aJaI_reverse;
clear mask_valid_aJaI_reverse;

% Linear indices (of the votings) into the accumulation array
lin2accum = sub2ind( size(grdmag), lin2accum_aI, lin2accum_aJ );

lin2accum_size = size( lin2accum );
lin2accum = reshape( lin2accum, [numel(lin2accum),1] );
clear lin2accum_aI lin2accum_aJ;

% Weights of the votings, currently using the gradient magnitudes
% but in fact any scheme can be used (application dependent)
weight4accum = ...
    repmat( double(grdmag(grdmasklin)) , [lin2accum_size(2),1] ) .*
...
    mask_valid_aJaI(:);
clear mask_valid_aJaI;

% Build the accumulation array using Matlab function 'accumarray'
accum = accumarray( lin2accum , weight4accum );
accum = [ accum ; zeros( numel(grdmag) - numel(accum) , 1 ) ];
accum = reshape( accum, size(grdmag) );

%%%%%%%%%% Locating local maxima in the accumulation array %%%%%%%%%%%%%%%

% Stop if no need to locate the center positions of circles
if ~func_compu_cen,
    return;
end
clear lin2accum weight4accum;

% Parameters to locate the local maxima in the accumulation array
% -- Segmentation of 'accum' before locating LM
prm_useaoi = true;
prm_aoithres_s = 2;
prm_aoiminsize = floor(min([ min(size(accum)) * 0.25, ...
    prm_r_range(2) * 1.5 ]));

% -- Filter for searching for local maxima
prm_fltrLM_s = 1.35;
prm_fltrLM_r = ceil( prm_fltrLM_R * 0.6 );
prm_fltrLM_npix = max([ 6, ceil((prm_fltrLM_R/2)^1.8) ]);

% -- Lower bound of the intensity of local maxima
prm_LM_LoBndRa = 0.2; % minimum ratio of LM to the max of 'accum'

% Smooth the accumulation array
fltr4accum = fltr4accum / sum(fltr4accum(:));

```

```

accum = filter2( fltr4accum, accum );

% Select a number of Areas-Of-Interest from the accumulation array
if prm_useaoi,
    % Threshold value for 'accum'
    prm_llm_thres1 = prm_grdthres * prm_aioithres_s;

    % Thresholding over the accumulation array
    accummask = ( accum > prm_llm_thres1 );

    % Segmentation over the mask
    [accumlabel, accum_nRgn] = bwlabel( accummask, 8 );

    % Select AOIs from segmented regions
    accumAOI = ones(0,4);
    for k = 1 : accum_nRgn,
        accumrgn_lin = find( accumlabel == k );
        [accumrgn_IdxE, accumrgn_IdxF] = ...
            ind2sub( size(accumlabel), accumrgn_lin );
        rgn_top = min( accumrgn_IdxE );
        rgn_bottom = max( accumrgn_IdxE );
        rgn_left = min( accumrgn_IdxF );
        rgn_right = max( accumrgn_IdxF );
        % The AOIs selected must satisfy a minimum size
        if ( (rgn_right - rgn_left + 1) >= prm_aioiminsize && ...
            (rgn_bottom - rgn_top + 1) >= prm_aioiminsize ),
            accumAOI = [ accumAOI; ...
                rgn_top, rgn_bottom, rgn_left, rgn_right ];
        end
    end
else
    % Whole accumulation array as the one AOI
    accumAOI = [1, size(accum,1), 1, size(accum,2)];
end

% Thresholding of 'accum' by a lower bound
prm_LM_LoBnd = max(accum(:)) * prm_LM_LoBndRa;

% Build the filter for searching for local maxima
fltr4LM = zeros(2 * prm_fltrLM_R + 1);

[mesh4fLM_x, mesh4fLM_y] = meshgrid(-prm_fltrLM_R : prm_fltrLM_R);
mesh4fLM_r = sqrt( mesh4fLM_x.^2 + mesh4fLM_y.^2 );
fltr4LM_mask = ...
    ( mesh4fLM_r > prm_fltrLM_r & mesh4fLM_r <= prm_fltrLM_R );
fltr4LM = fltr4LM - ...
    fltr4LM_mask * (prm_fltrLM_s / sum(fltr4LM_mask(:)));

if prm_fltrLM_R >= 4,
    fltr4LM_mask = ( mesh4fLM_r < (prm_fltrLM_r - 1) );
else
    fltr4LM_mask = ( mesh4fLM_r < prm_fltrLM_r );
end
fltr4LM = fltr4LM + fltr4LM_mask / sum(fltr4LM_mask(:));

% **** Debug code (begin)
if dbg_on,
    dbg_LMmask = zeros(size(accum));
end

```

```

% **** Debug code (end)

% For each of the AOIs selected, locate the local maxima
circen = zeros(0,2);
for k = 1 : size(accumAOI, 1),
    aoi = accumAOI(k,:);    % just for referencing convenience

    % Thresholding of 'accum' by a lower bound
    accumaoi_LBMask = ...
        ( accum(aoi(1):aoi(2), aoi(3):aoi(4)) > prm_LM_LoBnd );

    % Apply the local maxima filter
    candLM = conv2( accum(aoi(1):aoi(2), aoi(3):aoi(4)) , ...
        fltr4LM , 'same' );
    candLM_mask = ( candLM > 0 );

    % Clear the margins of 'candLM_mask'
    candLM_mask([1:prm_fltrLM_R, (end-prm_fltrLM_R+1):end], :) = 0;
    candLM_mask(:, [1:prm_fltrLM_R, (end-prm_fltrLM_R+1):end]) = 0;

    % **** Debug code (begin)
    if dbg_on,
        dbg_LMmask(aoi(1):aoi(2), aoi(3):aoi(4)) = ...
            dbg_LMmask(aoi(1):aoi(2), aoi(3):aoi(4)) + ...
            accumaoi_LBMask + 2 * candLM_mask;
    end
    % **** Debug code (end)

    % Group the local maxima candidates by adjacency, compute the
    % centroid position for each group and take that as the center
    % of one circle detected
    [candLM_label, candLM_nRgn] = bwlabel( candLM_mask, 8 );

    for ilabel = 1 : candLM_nRgn,
        % Indices (to current AOI) of the pixels in the group
        candgrp_masklin = find( candLM_label == ilabel );
        [candgrp_IdxI, candgrp_IdxJ] = ...
            ind2sub( size(candLM_label) , candgrp_masklin );

        % Indices (to 'accum') of the pixels in the group
        candgrp_IdxI = candgrp_IdxI + ( aoi(1) - 1 );
        candgrp_IdxJ = candgrp_IdxJ + ( aoi(3) - 1 );
        candgrp_idx2acm = ...
            sub2ind( size(accum) , candgrp_IdxI , candgrp_IdxJ );

        % Minimum number of qualified pixels in the group
        if sum(accumaoi_LBMask(candgrp_masklin)) < prm_fltrLM_npix,
            continue;
        end

        % Compute the centroid position
        candgrp_acmsum = sum( accum(candgrp_idx2acm) );
        cc_x = sum( candgrp_IdxJ .* accum(candgrp_idx2acm) ) / ...
            candgrp_acmsum;
        cc_y = sum( candgrp_IdxI .* accum(candgrp_idx2acm) ) / ...
            candgrp_acmsum;
        circen = [circen; cc_x, cc_y];
    end
end
end

```

```

% **** Debug code (begin)
if dbg_on,
    figure(dbg_bfigno); imagesc(dbg_LMmask); axis image;
    title('Generated map of local maxima');
    if size(accumAOI, 1) == 1,
        figure(dbg_bfigno+1);
        surf(candLM, 'EdgeColor', 'none'); axis ij;
        title('Accumulation array after local maximum filtering');
    end
end
% **** Debug code (end)

%%%%%%%%% Estimation of the Radii of Circles %%%%%%%%%%%%%%

% Stop if no need to estimate the radii of circles
if ~func_compu_radii,
    varargout{1} = circen;
    return;
end

% Parameters for the estimation of the radii of circles
fltr4SgnCv = [2 1 1];
fltr4SgnCv = fltr4SgnCv / sum(fltr4SgnCv);

% Find circle's radius using its signature curve
cirrad = zeros( size(circen,1), 1 );

for k = 1 : size(circen,1),
    % Neighborhood region of the circle for building the sgn. curve
    circen_round = round( circen(k,:) );
    SCvR_I0 = circen_round(2) - prm_r_range(2) - 1;
    if SCvR_I0 < 1,
        SCvR_I0 = 1;
    end
    SCvR_I1 = circen_round(2) + prm_r_range(2) + 1;
    if SCvR_I1 > size(grdx,1),
        SCvR_I1 = size(grdx,1);
    end
    SCvR_J0 = circen_round(1) - prm_r_range(2) - 1;
    if SCvR_J0 < 1,
        SCvR_J0 = 1;
    end
    SCvR_J1 = circen_round(1) + prm_r_range(2) + 1;
    if SCvR_J1 > size(grdx,2),
        SCvR_J1 = size(grdx,2);
    end

    % Build the sgn. curve
    SgnCvMat_dx = repmat( (SCvR_J0:SCvR_J1) - circen(k,1) , ...
        [SCvR_I1 - SCvR_I0 + 1 , 1] );
    SgnCvMat_dy = repmat( (SCvR_I0:SCvR_I1)' - circen(k,2) , ...
        [1 , SCvR_J1 - SCvR_J0 + 1] );
    SgnCvMat_r = sqrt( SgnCvMat_dx.^2 + SgnCvMat_dy.^2 );
    SgnCvMat_rpl = round(SgnCvMat_r) + 1;

    f4SgnCv = abs( ...

```

```

        double(grdx(SCvR_I0:SCvR_I1, SCvR_J0:SCvR_J1)) .* SgnCvMat_dx
+ ...
        double(grdy(SCvR_I0:SCvR_I1, SCvR_J0:SCvR_J1)) .* SgnCvMat_dy
...
    ) ./ SgnCvMat_r;
    SgnCv = accumarray( SgnCvMat_rpl(:) , f4SgnCv(:) );

    SgnCv_Cnt = accumarray( SgnCvMat_rpl(:) , ones(numel(f4SgnCv),1)
);
    SgnCv_Cnt = SgnCv_Cnt + (SgnCv_Cnt == 0);
    SgnCv = SgnCv ./ SgnCv_Cnt;

    % Suppress the undesired entries in the sgn. curve
    % -- Radii that correspond to short arcs
    SgnCv = SgnCv .* ( SgnCv_Cnt >= (pi/4 * [0:(numel(SgnCv_Cnt)-1)]') )
);
    % -- Radii that are out of the given range
    SgnCv( 1 : (round(prm_r_range(1))+1) ) = 0;
    SgnCv( (round(prm_r_range(2))+1) : end ) = 0;

    % Get rid of the zero radius entry in the array
    SgnCv = SgnCv(2:end);
    % Smooth the sgn. curve
    SgnCv = filtfilt( fltr4SgnCv , [1] , SgnCv );

    % Get the maximum value in the sgn. curve
    SgnCv_max = max(SgnCv);
    if SgnCv_max <= 0,
        cirrad(k) = 0;
        continue;
    end

    % Find the local maxima in sgn. curve by 1st order derivatives
    % -- Mark the ascending edges in the sgn. curve as 1s and
    % -- descending edges as 0s
    SgnCv_AscEdg = ( SgnCv(2:end) - SgnCv(1:(end-1)) ) > 0;
    % -- Mark the transition (ascending to descending) regions
    SgnCv_LMmask = [ 0; 0; SgnCv_AscEdg(1:(end-2)) ] &
(~SgnCv_AscEdg);
    SgnCv_LMmask = SgnCv_LMmask & [ SgnCv_LMmask(2:end) ; 0 ];

    % Incorporate the minimum value requirement
    SgnCv_LMmask = SgnCv_LMmask & ...
        ( SgnCv(1:(end-1)) >= (prm_multirad * SgnCv_max) );
    % Get the positions of the peaks
    SgnCv_LMPos = sort( find(SgnCv_LMmask) );

    % Save the detected radii
    if isempty(SgnCv_LMPos),
        cirrad(k) = 0;
    else
        cirrad(k) = SgnCv_LMPos(end);
        for i_radii = (length(SgnCv_LMPos) - 1) : -1 : 1,
            circen = [ circen; circen(k,:) ];
            cirrad = [ cirrad; SgnCv_LMPos(i_radii) ];
        end
    end
end

% Output

```



```

varargout{1} = circen;
varargout{2} = cirrad;
if nargin > 3,
    varargout{3} = dbg_LMmask;
end

```

## ***Příloha 2 – Program Pro detekci polohy částic***

```

clc
clear all
close all

h=imread('bb30_31.tif');
h=double(h(1:2048,1:2048)); %vezme pouze 2048 X 2048 hodnot (zadouci
pro FFT)
%% parametry pro rekonstrukci hologramu
lambda=532e-9; %vlnová délka
pixx=3.45e-6; %rozměr pixelu CCD
pixy=pixx;
Ur=1; % komplexní amplituda
vzdalenostd=31e-2; % spodni hranice vzdalenosti
vzdalenosth=32.5e-2; %horni hranice vzdalenosti

a=mean([vzdalenostd,vzdalenosth]); %stredni hodnota vzdalenosti

posun=0.5e-2; %krok
Circle=zeros(2,2); %pozice detekovanych castic
Radi=zeros(2,2); %velikost detekovanych castic
%% detekce castic z hologramu
for d=vzdalenostd:posun:vzdalenosth %nastaveni vzdáleností
    b=conv_rec(h,lambda,d,pixx,pixy,Ur,1); %funkce pro rekonstrukci
hologramu (pro popis zadej do prikazovy radky: help conv_rec)
    img=(abs(b)).^2;
    imwrite(img.*8,'test.bmp','bmp');
    img1=imread('test.bmp');

    if ((d>=a) & (d<a+posun))
        str=img1;
    end
    tic;
    [accum, circen, cirrad] = CircularHough_Grd(img1,[60,80],5,57,1);
    toc;
    %Vytvori pole obsahujici pozice a prumery jednotlivych nalezenych
bodů
    if (Circle(1,1)==0)
        Circle=circen;
        Radi=cirrad;
    else
        Circle=[Circle;circen];
        Radi=[Radi;cirrad];
    end
end

chyba=5; %odchylka
f=zeros(length(Circle),1);
for i=1:size(Circle,1)
    for j=1:size(Circle,1)

```

```

        if ((i~=j)&(Circle(i,1)+chyba>=Circle(j,1))&(Circle(i,1)-
chyba<=Circle(j,1))&(Circle(i,2)+chyba>=Circle(j,2))&(Circle(i,2)-
chyba<=Circle(j,2)))
            f(i)=f(i)+1;
        end
    end
end

r=Circle(find(f>=2),:); %vybere body ktere se opakují alespon 2x
q=Radi(find(f>=2)); %jejich velikosti
r1=r;
q1=q;
for i=1:size(r,1)
    for j=1:size(r,1)
        if ((r(i,1)+chyba>=r(j,1))&(r(i,1)-
chyba<=r(j,1))&(r(i,2)+chyba>=r(j,2))&(r(i,2)-chyba<=r(j,2))&(j>i))
            r1(j,:)=0;
            q1(j)=0;
        end
    end
end
r2=r(find(q1~=0),:);
q2=q(find(q1~=0));

figure(3);
imshow(str),
hold on;
plot(r2(:,1), r2(:,2), 'r+');
for k = 1 : size(r2, 1),
    DrawCircle(r2(k,1), r2(k,2), q2(k), 32, 'b-');
end
hold off;

%% Vyhledavani nejostrejsi castice

posun=0.2e-2;
vyber=80; %velikost ctverce okolo detekovaneho objektu
vzdalenostd=30e-2;
vzdalenosth=33e-2;

for l = 1 : size(r2, 1)
    h=imread('bb30_31.tif');
    h=double(h(1:2048,1:2048)); %vezme pouze 2048 X 2048 hodnot
    (zadouci pro FFT)
    j=0;
    x=vzdalenostd;

    if (r2(l,1)-vyber)<0
        xd(l)=1;
    else
        xd(l)=(r2(l,1)-vyber);
    end

    if (r2(l,1)+vyber)>2048
        xh(l)=2048;
    else

```

```

        xh(1)=(r2(1,1)+vyber);
    end

    if (r2(1,2)-vyber)<0
        yd(1)=1;
    else
        yd(1)=(r2(1,2)-vyber);
    end

    if (r2(1,2)+vyber)>2048
        yh(1)=2048;
    else
        yh(1)=(r2(1,2)+vyber);
    end

    for d=vzdalenostd:posun:vzdalenosth %nastaveni vzdáleností
        b=conv_rec(h,lambda,d,pixx,pixy,Ur,1); %funkce pro
        rekonstrukci hologramu (pro popis zadej do prikazovy radky: help
        conv_rec)
        I=(abs(b)).^2;

        I=double(I(yd(1):yh(1),xd(1):xh(1)));
        j=j+1;
        B(j)=x;

        Fx = gradient(I); % gradient obrazku
        C(j)=sum(sum(Fx));
        x=x+posun;
    end

    [maximum,pozice]=max(abs(C)); %vrati pozici nejvetsiho prvku
    z(1)=B(pozice);
    disp(['Nejostrejsi obrazek je cislo ',num2str(pozice+1),' ve
    vzdalenosti ',num2str(B(pozice))]);
end
%% Zobrazeni 3D modelu a vypsani souradnic jednotlivych castic

x=(z(:)-vzdalenostd);
x=x.*100;
souradnice=[r2.*pixy*1000 x]
figure(4);
maxim= ceil (max(max(souradnice)))
scatter3(souradnice(:,1),souradnice(:,2),souradnice(:,3),'filled');
title('Souřadnice detekovaných částic');
Xlabel('X [mm]')
Ylabel('Y [mm]')
Zlabel('Z [mm]')
axis equal;
axis([0 maxim,0 maxim,0 maxim])
grid on

```